

Home Computing WEEKLY

An Argus Specialist Publication

**TI99-4A
collection**

ocean



Bombs away – and you're in the pilot's seat



In my Bomber program you are the pilot of a crippled jet and the only way to land safely is to destroy the towers to clear a landing strip.

You are given the choice of three skill levels, each giving you a different number of bombs.

When you have chosen the game screen will be set up with your bomber at the top of the screen and a set of towers of random heights at the bottom.

Press any key to start the bomber moving. Engine trouble means you are losing height.

To destroy the towers, press key B to release your bombs. The bombs travel at an angle of 45 degrees and destroy buildings if they hit the roofs.

On levels two and three you can only release two bombs for every pass across the screen. If you hit a building your bomber will be destroyed.

The game can then be replayed or you can end.

How it runs:

10-110 define characters
120-270 set colours, level, variables

Here's two versions of the classic arcade game, Bomber.

First John Power presents his program for the TI 99/4A and then Mark Bowyer describes his version for both models of the BBC micro

280-450 set layout of screen and bombs counter

460-630 set random height, pattern of towers and prints them

640-910 move bomber, check if B is pressed, updates score
910-1200 move bomber and bomb, check if bomb hits tower
1210-1270 erase tower, sounds explosion

1280-1330 routine to print score
1340-1570 bomber crashed, prints info on score, option to run or end

1580-1830 mission successful, then as above

1840-2240 starting sub-routine, set variables according to level, print text

Apart from some unusual graphics commands, TI Basic is mostly standard. The Texas machine will not accept multiple assignments, nor multiple line statements. Room could be saved on other machines which have these facilities, especially in the case of long variable lists.

The CALL CHAR statement re-defines characters over the ASCII range 32-159. The characters take the form of an

eight by eight grid square split vertically into two four by eight halves. Therefore one hexadecimal number refers to one pattern in one of the four by eight blocks. The character is defined left to right, from the top.

CALL HCHAR is used to position the character on the 32 by 24 screen, and repeat it horizontally. This can be replaced by a poke to the screen, or similar statement on other machines.

CALL VCHAR repeats a character vertically. CALL GCHAR returns the ASCII value of the character present on the screen at the given co-ordinates. This can be used to detect collisions, and can be replaced by peek on other machines.

CALL CLEAR clears the screen, CALL SCREEN sets the screen colour, and CALL COLOR sets character color, the arguments being character set number (on the TI ASCII codes are split into 16 blocks of eight codes) foreground color, background color. CALL SOUND speaks for itself, and is easily replaced by a similar command on another machine.

BBC AND TI99/4A PROGRAMS

Variables:

OBR — Bombs (original)
BMS — Bombs
RW — Row (on screen)
CL — Column (on screen)
CHO — Original bomber position
SH — Shots made per line
RS — Random tower pattern
TW — ASCII value of character

RH — Random height of towers
COL — screen positions of towers
SL — Skill level
Z — Row of printed message
V — Col of printed message
TC — Variable holding ASCII char value found by GCHAR
TH — same as above
CT — Col position of bomb
RT — Row position of bomb

Bomber for the TI 99/4A

```
10 REM ***BOMBER J.POWER 15/1/83***
20 CALL CHAR(96,"55AA55AA55AA55AA")
30 CALL CHAR(97,"0F0F0F0F0F0F0F0F")
40 CALL CHAR(98,"FF00FF00FF00FF00")
50 CALL CHAR(104,"209098FE7F1B1020")
60 CALL CHAR(105,"0000207020000000")
70 CALL CHAR(99,"00000000000000FF")
80 CALL CHAR(112,"000000001B3C7EFF")
90 CALL CHAR(100,"00FF000000000000")
100 CALL CHAR(107,"B29254EE5492B200")
110 CALL CHAR(120,"FFFFFFFFFFFFFFFF")
120 CALL COLOR(9,2,15)
130 CALL COLOR(12,15,6)
140 CALL COLOR(10,2,6)
150 CALL COLOR(11,5,6)
160 CALL CLEAR
170 REM **STARTING ROUTINE**
180 GOSUB 1840
190 REM **RETURN FROM STARTING ROUTINE-WITH DATA
200 CALL CLEAR
210 FOR QQ=3 TO 8
220 CALL COLOR(QQ,2,15)
230 NEXT QQ
240 OBR=BMS
250 RW=CHO
260 CL=2
270 SH=0
280 FOR X=1 TO 3
290 CALL HCHAR(X,1,120,32)
300 NEXT X
310 CALL HCHAR(23,1,120,32)
320 CALL HCHAR(24,1,120,32)
330 CALL SCREEN(6)
340 M$="BOMBER"
350 Z=2
360 V=6
370 GOSUB 1290
380 CALL HCHAR(1,7,99,6)
390 CALL HCHAR(3,7,100,6)
400 M$="BOMBS="+STR$(BMS)
410 Z=2
420 V=18
430 GOSUB 1290
440 CALL HCHAR(1,19,99,8)
450 CALL HCHAR(3,19,100,8)
460 REM **SET UP TOWERS**
470 COL=7
480 FOR RTD=1 TO 20
490 RANDOMIZE
500 RS=INT(RND*3)+1
510 IF RS=1 THEN 520 ELSE 540
520 TW=96
530 GOTO 590
540 IF RS=2 THEN 550 ELSE 570
550 TW=97
560 GOTO 590
570 IF RS=3 THEN 580 ELSE 500
580 TW=98
590 RH=INT((HL-LL+1)*RND)+LL
600 CALL VCHAR(22-RH,COL,112)
610 CALL VCHAR(23-RH,COL,TW,RH)
620 COL=COL+1
630 NEXT RTD
640 REM **BOMBER MOVEMENT**
650 CALL KEY(0,K,S)
660 CALL HCHAR(RW,CL,104)
670 CALL HCHAR(RW,CL,32)
```

```
680 IF S=0 THEN 650
690 CALL HCHAR(RW,CL,104)
700 CALL KEY(0,K,S)
710 IF S=1 THEN 720 ELSE 760
720 IF K<>66 THEN 760
730 IF (SL=2)+(SL=3) THEN 740 ELSE 750
740 IF SH=2 THEN 760 ELSE 750
750 IF CL>=32 THEN 760 ELSE 910
760 CALL HCHAR(RW,CL,32)
770 CL=CL+1
780 IF CL>32 THEN 790 ELSE 870
790 M$=STR$(BMS)+"X"
800 Z=2
810 V=24
820 GOSUB 1290
830 SH=0
840 CL=1
850 RW=RW+1
860 IF RW=23 THEN 1580
870 CALL GCHAR(RW,CL,TC)
880 IF TC=112 THEN 1340 ELSE 890
890 CALL HCHAR(RW,CL,104)
900 GOTO 700
910 REM **BOMB AND BOMBER MOVEMENT**
920 RT=RW+1
930 CT=CL+1
940 IF BMS=0 THEN 700
950 BMS=BMS-1
960 SH=SH+1
970 CALL GCHAR(RT,CT,TH)
980 IF TH=112 THEN 1210 ELSE 990
990 CALL HCHAR(RT,CT,105)
1000 CALL SOUND(-100,-5,2)
1010 CALL HCHAR(RW,CL,32)
1020 CL=CL+1
1030 IF CL>32 THEN 1040 ELSE 1070
1040 CL=1
1050 SH=0
1060 RW=RW+1
1070 CALL GCHAR(RW,CL,TC)
1080 IF TC=112 THEN 1340 ELSE 1090
1090 CALL HCHAR(RW,CL,104)
1100 CALL HCHAR(RT,CT,32)
1110 CT=CT+1
1120 RT=RT+1
1130 IF RT>=22 THEN 700 ELSE 1140
1140 IF CT>32 THEN 700
1150 CALL GCHAR(RT,CT,TH)
1160 IF TH=112 THEN 1210 ELSE 1170
1170 IF (TH=96)+(TH=97)+(TH=98) THEN 700
1180 CALL HCHAR(RT,CT,105)
1190 CALL SOUND(-50,2000,20)
1200 GOTO 1010
1210 REM **TOWER ERASE**
1220 CALL HCHAR(RT,CT,107)
1230 CALL SOUND(-750,110,2,-6,0,192,2)
1240 CALL VCHAR(RT,CT,32,23-RT)
1250 Z=2
1260 V=24
1270 GOTO 700
1280 REM ** "PRINT AT" ROUTINE**
1290 FOR LOS=1 TO LEN(M$)
1300 CODE=ASC(SEG$(M$,LOS,1))
1310 CALL HCHAR(Z,V+LOS,CODE)
1320 NEXT LOS
1330 RETURN
1340 REM *****BOMBER CRASHED*****
1350 CALL HCHAR(RW,CL,107)
1360 CALL SOUND(-1250,196,2,-7,0,110,2)
1370 FOR X=1 TO 250
1380 NEXT X
1390 FOR QQ=3 TO 8
1400 CALL COLOR(QQ,2,8)
1410 NEXT QQ
1420 CALL CLEAR
1430 CALL SCREEN(8)
1440 GOSUB 2170
1450 PRINT TAB(3);"YOUR BOMBER HAS CRASHED.":
1460 PRINT TAB(5);"YOU HAD";BMS;"BOMBS LEFT-":
1470 PRINT TAB(4);"OUT OF YOUR ORIGINAL";OBR
1480 PRINT :
1490 GOSUB 2200
1500 PRINT :
```


BBC AND TI99/4A PROGRAMS

```

1510 CALL KEY(0,K,S)
1520 IF S=0 THEN 1510
1530 IF K<>80 THEN 1550
1540 GOTO 180
1550 IF K<>69 THEN 1510
1560 CALL CLEAR
1570 END
1580 REM ***MISSION SUCCESSFUL***
1590 CALL SOUND(200,370,2,440,2,294,2)
1600 CALL SOUND(600,392,2,494,2,294,2)
1610 CALL SOUND(200,391,2,494,2,292,2)
1620 CALL SOUND(800,294,2,440,2,370,2)
1630 FOR QQ=3 TO 8
1640 CALL COLOR(QQ,2,8)
1650 NEXT QQ
1660 CALL CLEAR
1670 CALL SCREEN(8)
1680 GOSUB 2170
1690 OBRZ=OBR
1700 BL=OBR-BMS
1710 PRINT "CONGRATULATIONS, YOUR MISSION":
1720 PRINT TAB(9); "WAS SUCCESSFUL.":
1730 PRINT TAB(7); "YOU USED"; BL; "BOMBS.":
1740 PRINT TAB(5); "FROM YOUR ORIGINAL"; OBRZ
1750 PRINT :
1760 GOSUB 2200
1770 CALL KEY(0,K,S)
1780 IF S=0 THEN 1770
1790 IF K<>80 THEN 1810
1800 GOTO 180
1810 IF K<>69 THEN 1770
1820 CALL CLEAR
1830 END

```

```

1840 REM **STARTING ROUTINE**
1850 CALL CLEAR
1870 PRINT "PRESS:":
1880 PRINT TAB(5); "1 FOR LEVEL 1-30 BOMBS":
1890 PRINT TAB(5); "2 FOR LEVEL 2-24 BOMBS":
1900 PRINT TAB(5); "3 FOR LEVEL 3-22 BOMBS":
1930 CALL KEY(0,K,S)
1940 IF S=0 THEN 1930
1950 IF K<>49 THEN 2020
1960 SL=1
1970 HL=6
1980 LL=4
1990 BMS=30
2000 CHO=4
2010 GOTO 2150
2020 IF K<>50 THEN 2090
2030 SL=2
2040 HL=7
2050 LL=4
2060 BMS=24
2070 CHO=4
2080 GOTO 2150
2090 IF K<>51 THEN 1930
2100 SL=3
2110 HL=7
2120 LL=3
2130 BMS=22
2140 CHO=6
2150 RETURN
2210 PRINT "PRESS: P TO PLAY      "
2220 PRINT "E TO END.          "
2230 RETURN

```

Bomber for the BBC micro is also based on the idea of destroying a skyline to land your aircraft.

The program displays the instructions in Mode 7 and plays the game in Mode 5 using programmable characters.

Here's an explanation of some of the commands used in the program:

*FX 15,0 in line 120 clears the contents of the keyboard buffer.

*FX 4,1 allows the cursor keys to give their codes rather

than their specialist functions. The codes used are: hex 8A down arrow, hex 8B up arrow, decimal 70 F key.

*FX 4,0 returns the cursor keys to their normal functions.

The SOUND commands in lines 70 and 280 create short explosion-type noises, the one in 280 being slightly louder and longer than the one in 70.

*FX 11,14 sets the auto repeat speed.

*FX 12 returns all keyboard associated attributes to normal.

VDU 23 in lines 610-650 define character codes as follows: 250 checkered black, 251 tail section of bomber, 252 centre sec-

tion, 253 front section of bomber, 254 the bomb.

DEF FNS(X,Y)...in line 760 returns the first byte, of eight, of a character at co-ordinates (X,Y). It is used in line 70 to detect whether the bomber has hit the skyline block and in 730 to detect whether the bomb has hit a skyline block. This function works only in Mode 7.

The PROC sub-routines:

PROCINSTRUCTIONS prints instructions in Mode 7.

PROCINIT initialises variables and prints a randomly generated skyline.

The movement of the

bomber takes place in lines 60-220.

PROCBOMBINIT initialises the bomb position and prints it.

PROCBOMB erases the bomb, checks to see if a skyline row has been completely destroyed or checks to see whether a bomb has absorbed four blocks of skyline.

PROCLANDED prints a 'landed' message in yellow and passes control to the 'Another game' routine in lines 350-440.

PROCCRASH does the same as PROCLANDED, except it prints a crash message instead along with an explosion covering the bomber.

BBC micro version of the Bomber

```

10 MODE 7
20 PROCINSTRUCTIONS
30 MODE 5
40 PROCINIT
50 FOR X=0 TO 19
60   X=X+1
70   IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN L=1:GOTO 80:IF L=1
80   PRINTTAB(12,25)CHR$(250)
90   PRINTTAB(12,25)CHR$(251)
100  PRINTTAB(12,25)CHR$(253)
110  IF L=0 THEN PROCBOMB:GOTO 350
120  IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN PROCBOMBINIT
130  IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN PROCBOMB
140  IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN PROCBOMB
150  IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN PROCBOMB
160  IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN PROCBOMB
170  IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN PROCBOMB
180  IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN PROCBOMB
190  IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN PROCBOMB
200  NEXT X
210  GOTO 50
220  DEF PROCCLASH
230  COLOUR 2
240  PRINTTAB(12,25)CHR$(250)
250  ENDPROC
260  DEF PROCBOMB
270  SOUND 600,392,2,494,2,294,2
280  FOR Q=0 TO 120
290    IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN PROCBOMB
300    IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN PROCBOMB
310    IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN PROCBOMB
320    COLOUR 2
330    PRINTTAB(12,25)CHR$(250)
340  ENDPROC
350  FOR L=0 TO 20:GOTO 360
360  COLOUR 2
370  PRINTTAB(12,25)CHR$(250)
380  IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN PROCBOMB
390  IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN PROCBOMB
400  IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN PROCBOMB
410  IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN PROCBOMB

```

```

420  GOTO 390
430  *FX 4,0
440  MODE 7:END
450  DEF PROCBOMB
460  PRINTTAB(12,25)CHR$(250)
470  IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN PROCBOMB
480  IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN PROCBOMB
490  IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN PROCBOMB
500  DEF PROCINSTRUCTIONS
510  *FX 4,1
520  PRINTTAB(12,25)CHR$(250)
530  PRINTTAB(12,25)CHR$(251)
540  PRINTTAB(12,25)CHR$(253)
550  PRINTTAB(12,25)CHR$(254)
560  PRINTTAB(12,25)CHR$(255)
570  DEF PROCINIT
580  *FX 11,14
590  VDU 23:PRINTTAB(12,25)CHR$(250)
600  L=0:GOTO 610
610  VDU 23:PRINTTAB(12,25)CHR$(250)
620  VDU 23:PRINTTAB(12,25)CHR$(251)
630  VDU 23:PRINTTAB(12,25)CHR$(253)
640  VDU 23:PRINTTAB(12,25)CHR$(254)
650  VDU 23:PRINTTAB(12,25)CHR$(255)
660  FOR X=0 TO 19
670    IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN PROCBOMB
680    IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN PROCBOMB
690    IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN PROCBOMB
700  ENDPROC
710  DEF PROCBOMBINIT
720  *FX 4,1
730  IF (FNS(X,2)=0 OR FNS(X,3)=0) THEN PROCBOMBINIT
740  PRINTTAB(12,25)CHR$(250)
750  ENDPROC
760  DEF FNS(X,Y)=FNS(X,Y)

```


How does your TI screen seem?

Standard Basic treats the screen of the TI 99/4A computer as having 32 by 24 characters. But you can get higher resolution by dividing each character position into a number of pixels.

I'll be giving a series of Basic sub-routines to allow you to define and manipulate these pixels.

If we divide each character position into four pixels then, since each pixel may be set or unset, we need 16 (2^4) different patterns, each of four pixels.

We can represent each of these patterns as a four-bit



Dreaming of higher screen resolution with your TI 99/4A micro? David Gray shows you how

numeral; one bit for each pixel as shown in the illustration. The patterns and their associated binary numerals are shown also.

Using four pixels per character position gives us a

screen consisting of 64 by 48 pixels. However, since Basic only allows us to write characters we must define one character for each pattern and to set or unset a pixel we must change the com-

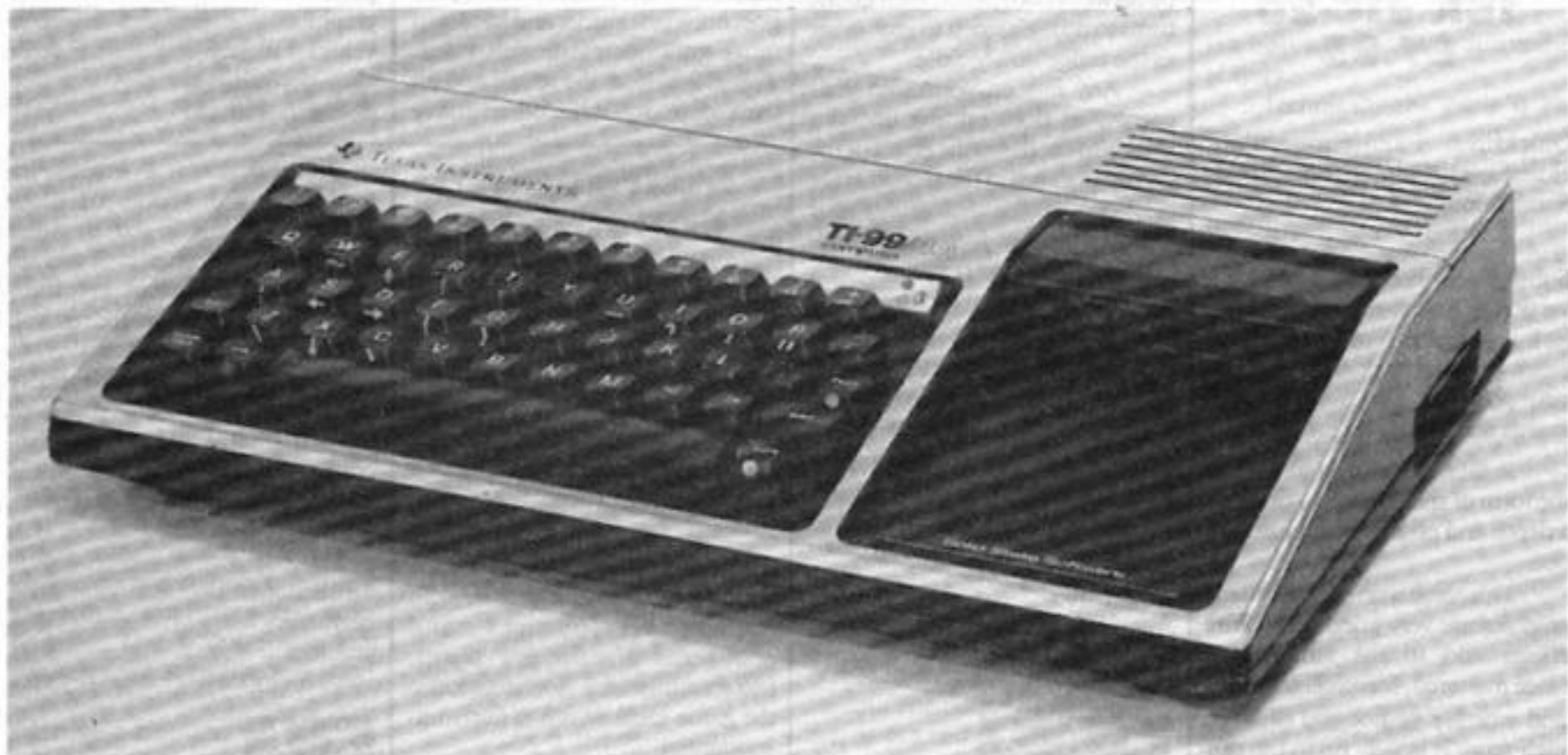
plete character containing that pixel.

Our method of numbering patterns makes the determination of the new pattern very simple.

If we start with a pattern p ($0 \dots 15$) and wish to set pixel n ($0 \dots 3$), we get pattern $p + 2^n$ (providing pixel n is not already set).

Similarly, unsetting pixel n of pattern p gives pattern $p - 2^n$ (providing pixel n is already set). For example, setting pixel 2 in pattern 8 gives pattern $8 + 2^2 = 12$.

On the TI 99/4A computer,



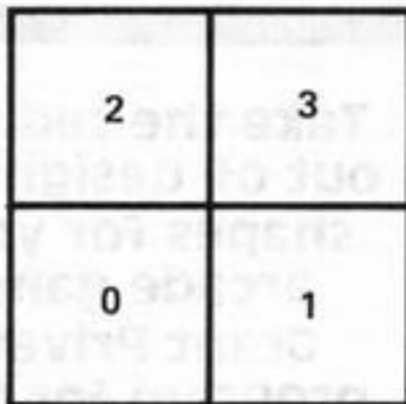
PROGRAMMING

the character at the top left-hand corner of the screen has co-ordinates (1,1). Here we assume that the Xco-ordinates run horizontally from 1 to 32 and that the Yco-ordinates run vertically from 1 to 24.

For this article we wish to

treat the pixel in bottom left-hand corner as (0,0), so pixel (0,0) will be pixel 0 of character position (1,24).

To access any pixel we must determine which character position it belongs to and which pixel it is of that character position.



1 CHARACTER

How one character is divided into four pixels. Each character can be represented as a four-bit numeral

```
10 DEF DIV2(X)=INT(X/2)
20 DEF MOD2(X)=X-INT(X/2)*2
30 REM A SIMPLE TEST PROGRAM
40 REM FIRST FILL PART OF THE SCREEN PIXEL BY PIXEL
45 GOSUB 500
50 FOR PLOT_X=30 TO 35
60 FOR PLOT_Y=25 TO 30
70 GOSUB 1000
80 NEXT PLOT_Y
90 NEXT PLOT_X
100 REM NOW REMOVE SOME OF THE PIXELS
110 FOR PLOT_Y=30 TO 25 STEP -1
120 FOR PLOT_X=33 TO 30 STEP -1
130 GOSUB 1100
140 NEXT PLOT_X
150 NEXT PLOT_Y
```

```
160 STOP
500 REM S/R INITIALISE GRAPHICS CHARACTERS
510 DIM PLOT_S$(3)
520 PLOT_S$(0)="00000000"
530 PLOT_S$(1)="FOFOFOFO"
540 PLOT_S$(2)="OFOFOFOF"
550 PLOT_S$(3)="FFFFFFF"
560 PLOT_CH=128
570 FOR PLOT_I=0 TO 3
580 FOR PLOT_J=0 TO 3
590 CALL CHAR(PLOT_CH,PLOT_S$(PLOT_I)&PLOT_S$(PLOT_J))
600 PLOT_CH=PLOT_CH+1
610 NEXT PLOT_J
620 NEXT PLOT_I
630 CALL COLOR(13,13,16)
640 CALL COLOR(14,13,16)
650 CALL HCHAR(1,1,128,24*32)
660 RETURN
```

```
1000 REM S/R TO PLOT POINT <PLOT_X,PLOT_Y>
1010 GOSUB 2000
1020 IF PLOT_FLAG THEN 1060
1030 GOSUB 2500
1040 IF PLOT_FLAG THEN 1060
1050 CALL HCHAR(PLOT_Y1,PLOT_X1,PLOT_CH+2^PLOT_BIT)
1060 RETURN
1100 REM S/R TO UNPLOT POINT <PLOT_X,PLOT_Y>
1110 GOSUB 2000
1120 IF PLOT_FLAG THEN 1160
1130 GOSUB 2500
1140 IF PLOT_FLAG+1 THEN 1160
1150 CALL HCHAR(PLOT_Y1,PLOT_X1,PLOT_CH-2^PLOT_BIT)
1160 RETURN
```

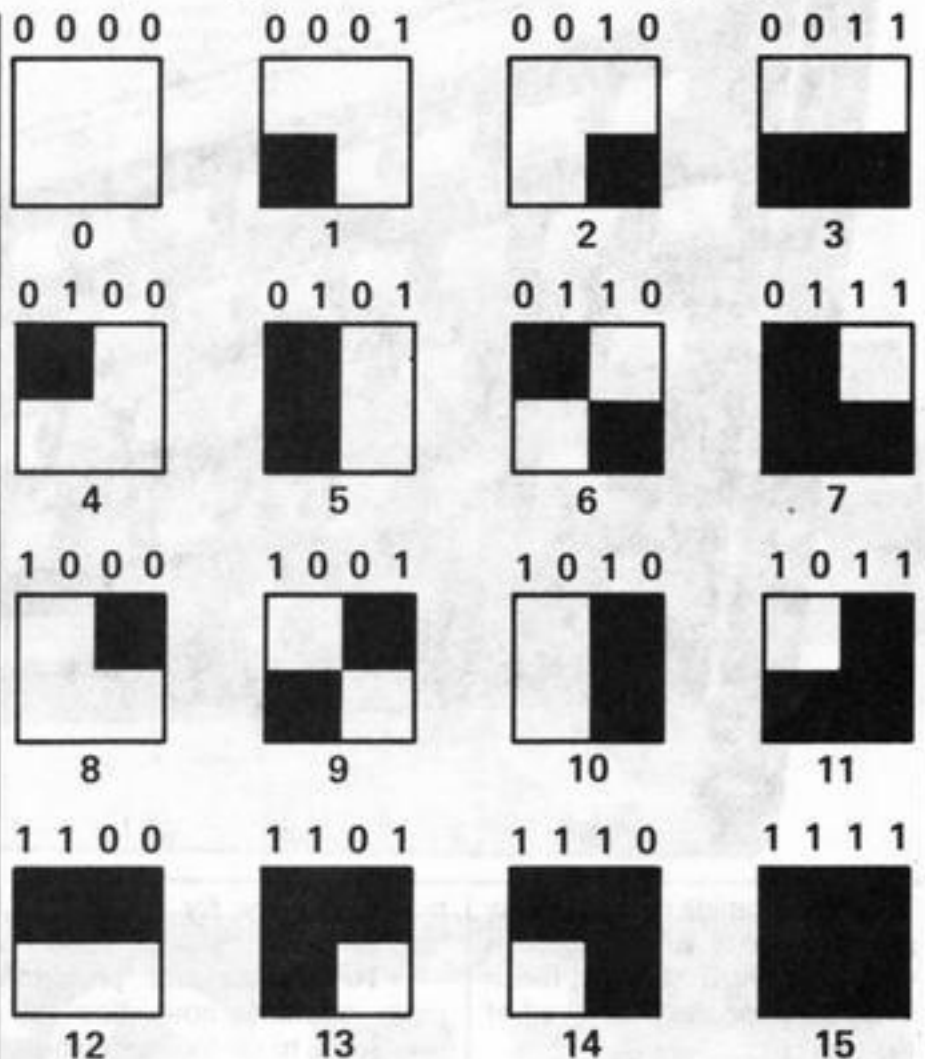
```
2000 REM S/R TO DETERMINE CHARACTER AND PIXEL
2010 PLOT_FLAG=-1
2020 IF (PLOT_X<0)+(PLOT_X>63)+(PLOT_Y<0)+(PLOT_Y>47) THEN 2080
2030 PLOT_X1=DIV2(PLOT_X)+1
2040 PLOT_Y1=24-DIV2(PLOT_Y)
2050 PLOT_BIT=MOD2(PLOT_X)+2*MOD2(PLOT_Y)
2060 CALL GCHAR(PLOT_Y1,PLOT_X1,PLOT_CH)
2070 PLOT_FLAG=0
2080 RETURN
2500 REM S/R TO SEE IF PIXEL ALREADY PLOTTED
2510 PLOT_FLAG=MOD2(INT((PLOT_CH-128)/(2^PLOT_BIT)))=1
2520 RETURN
```

Given pixel (x,y) then the corresponding character position is given by $(1 + x \text{ DIV } 2, 24 - y \text{ DIV } 2)$ where $x \text{ DIV } 2$ is the integer result of dividing x by 2.

The pixel of the determined character position is given by $(x \text{ MOD } 2) + 2 * (Y \text{ MOD } 2)$

where $x \text{ MOD } 2$ is the remainder of dividing x by 2.

Using the above scheme we can write three Basic sub-routines to produce higher resolution — one to initialise a character for each pattern, one to plot a pixel and one to unplot a pixel.



Sixteen different shapes can be created by using four pixels in one character pattern. Their binary numbers are shown above each

How it works

10,20 Definitions of two commonly used functions.
30-160 A simple test program
500-660 A subroutine to initialise the characters representing the various patterns. These patterns are represented by the characters 128 to 143
650 Sets all character positions to the blank pattern
1000- This subroutine plots the point (PLOT - X, PLOT - Y) on the screen. It first makes sure that the point is valid by calling subroutine 2000 and then ensures that the point hasn't already been plotted by calling subroutine 2500. Line 1050 uses the formula $p + 2^n$
1100- Similar to the subroutine 1000 but for unplotting the point
1160 (PLOT - X, PLOT - Y)

2000- This subroutine determines if the point PLOT - X, PLOT - Y is actually on the screen (see Line 2020). If the point is not on the screen then PLOT-FLAG is left as true (-1). If the point is on the screen then PLOT-FLAG is left as false (0) and the following variables are set: -(PLOT - X1, PLOT - Y1) are set to the coordinates of the character position containing the pixel. PLOT - BIT is set to the pixel number within the character position. PLOT - CH is set to the current character in the character position
2500- This subroutine returns the value false if the pixel PLOT_BIT of the character PLOT - CH is not set and true if it is set.

Playing around with pixels — on screen

Creating user-defined characters is a tricky task. The TI-99/4A has a special routine, called CHAR, to deal with it. Even so, to define each 8 x 8 pixel character, you have to work out, then type in, a 16-digit hexadecimal string — and after all that you may well find that the end result that appears on screen is not quite what you intended!

My program, for the TI-99/4A, allows you to design up to four characters on screen. Enlarged version of the characters can be manipulated on a large 16 x 16 grid, and moved around it by using the arrow keys. Individual pixels can be set or removed by using ENTER.

As well as the large grid, the program also displays normal-sized representations of the characters and their hexadecimal string representations. These are only updated when the user moves to another character or when the space bar is pressed.

At the start of the program, users are given the option of supplying initial values for the characters. Only if they wish to do this must a 16-digit hexadecimal number be supplied.

To make the program as foolproof as possible, all user input is checked and any attempt to move the cursor off the 16 x 16 grid is rejected.

Going through the program line by line, this is how it works:



Finding user-defined characters hard to handle on the TI-99/4A? David Gray shows you how to tame them

150	dimensions the arrays used in the program. ST\$ holds the strings representing the four characters being defined. MSG\$ holds parts of messages to be displayed during initialisation. REP (3,15) holds the 16 hexadecimal digits making up the representations for the four characters.	480	moves cursor right.
160-240	define some functions used in the rest of the program. DIVn (X) produces the integer division of X by n and MODn (X) produces the remainder.	510	moves cursor down.
250-270	calls subroutines to carry out initialisation.	540	moves cursor up.
280-310	a loop to output the string representation and initial display for each of the four characters.	570	sets or unsets the current pixel (indicated by CURX and CURY) of the current character (STI).
320	makes a sound	590	Invalid input, so make a noise.
330	STI will always contain the number (0 to 3) of the current character being defined. This line sets STI to represent, the top left hand character.	The remainder of the program is a collection of subroutines.	
340	CHANGED will indicate whether the current character (STI) has been changed since the last time its string representation and normal-sized representation were displayed. FALSE (0) implies no change and TRUE (-1) implies a change.	620-880	clear the screen and set up the user defined characters and colours used in the program. character codes 124 to 127 are used to define the four characters being created.
350-360	CRUX and CURY indicate the coordinates of the cursor within the current character (STI). (0,0) indicates the top left hand corner of the character.	890-1110	clear the screen and set up the empty display including the 16 x 16 grid.
370	puts the cursor on the screen.	1120-1220	display the string representation (ST\$(STI)) for the character STI at the appropriate position on the screen.
380-390	wait for user input.	1230-1540	ask whether the user wishes to supply initial values for the four characters. If so, this subroutine inputs the values and calls the subroutine at 2760 to check that they are valid.
400	removes the cursor from the screen.	1550-1580	determine the absolute position of the cursor on the screen by using the character number STI and the position within that character given by CURX and CURY.
410-600	perform some action depending on the value of the input (KEY) and then goto 370.	1590-1660	position the cursor on the screen by finding its absolute position using the subroutine at 1550. The type of cursor plotted depends on whether the cursor is at a set or unset pixel. This is determined by calling GCHAR.
420	Space bar has been pressed, so updates the display if the character (STI) has been changed.		
450	moves cursor left.		

TI-99/4A PROGRAMMING

- 1670-1740 removes the cursor and replaces it by the appropriate character to represent a set or unset pixel.
 1750-1850 move cursor up.
 1960-1960 move cursor down.
 1970-2070 move cursor right.
 2080-2180 move cursor left.

These four subroutines move the cursor as directed by the user. If any attempt is made to move the cursor off the 16 x 16 grid, a sound is made (by calling the subroutine at 2190) and no action is taken.

If the move takes the cursor to another character, then the display is updated by calling the subroutine at 2920. Otherwise CURX or CURY is updated as appropriate. Note: at this point the cursor is not actually displayed on the screen, but when it is replaced STI, CURX and CURY will indicate its correct position.

```

100 REM Character Generator
110 REM . A.E.Gray
120 REM 1983
130 REM
140 REM MAIN PROG
150 DIM ST$(3),MSG$(3),REP(3,15)
160 REM Some common functions.
170 DEF MOD2(X)=X-2*INT(X/2)
180 DEF DIV2(X)=INT(X/2)
190 DEF MOD4(X)=X-4*INT(X/4)
200 DEF DIV4(X)=INT(X/4)
210 DEF MOD8(X)=X-8*INT(X/8)
220 DEF DIV8(X)=INT(X/8)
230 DEF DIV16(X)=INT(X/16)
240 DEF MOD16(X)=X-16*INT(X/16)
250 GOSUB 620
260 GOSUB 1230
270 GOSUB 890
280 FOR STI=0 TO 3
290 GOSUB 1120
300 GOSUB 2260
310 NEXT STI
320 GOSUB 2190
330 STI=0
340 CHANGED=0
350 CURX=0
360 CURY=0
370 GOSUB 1590
380 CALL KEY(3,KEY,STATUS)
390 IF STATUS=0 THEN 380
400 GOSUB 1670
410 IF KEY<>32 THEN 440
420 GOSUB 2920
430 GOTO 370
440 IF (KEY<>8)*(KEY<>83)*(KEY<>115) THEN 470
450 GOSUB 2080
460 GOTO 370
470 IF (KEY<>9)*(KEY<>68)*(KEY<>100) THEN 500
480 GOSUB 1970
490 GOTO 370
500 IF (KEY<>10)*(KEY<>88)*(KEY<>120) THEN 530
510 GOSUB 1860
520 GOTO 370
530 IF (KEY<>11)*(KEY<>69)*(KEY<>101) THEN 560
540 GOSUB 1750
550 GOTO 370
560 IF KEY<>13 THEN 590
570 GOSUB 2530
580 GOTO 370
590 GOSUB 2190
600 GOTO 370
610 STOP
620 REM INITIALISE CHARACTERS FOR PROGRAM DISPLAY
630 CALL CLEAR
640 CALL CHAR(112,"00000000000000FF")
650 CALL CHAR(113,"FF")
660 CALL CHAR(114,"7EB19DA1A19DB17E")
670 CALL CHAR(115,"0")
680 CALL CHAR(116,"0101010101010101")
690 CALL CHAR(117,"8080808080808080")
700 CALL COLOR(11,2,1)
710 CALL CHAR(120,"FFFFFFFFFFFFFFFF")
720 CALL CHAR(121,"00001B3C3C1B")
730 CALL CHAR(122,"FFFE7C3C3E7FFF")
740 CALL CHAR(123,"0")
    
```

- 2460-2520 extract the DIGITth digit from the string S\$ and turn it into an integer in the range 0 to 15.
 2530-2750 change the current pixel when the user presses ENTER. The change is recorded by setting CHANGED to TRUE (-1). The subroutine then updates the display by changing the character at the cursor position. It then replaces the value of REP with the current character STI to reflect the change. At this point the string representation for the character STI remains unchanged.
 2760-2910 check a string (S\$) input by the user and report any errors found. If the string is correct a value of TRUE (-1) is returned in OK; otherwise a value of FALSE (0) is returned in OK.
 2920-3110 update the string representation and the normal character representation of the character STI, if it has changed. The string representation is updated both internally (ST\$(STI)) and on the display.

```

750 CALL CHAR(124,ST$(0))
760 CALL CHAR(125,ST$(1))
770 CALL CHAR(126,ST$(2))
780 CALL CHAR(127,ST$(3))
790 CALL COLOR(12,2,16)
800 CALL CHAR(128,"01010101010101FF")
810 CALL COLOR(13,2,6)
820 CALL CHAR(136,"FF01010101010101")
830 CALL COLOR(14,2,10)
840 CALL CHAR(144,"80808080808080FF")
850 CALL COLOR(15,2,12)
860 CALL CHAR(152,"FF80808080808080")
870 CALL COLOR(16,2,15)
880 RETURN
890 REM SET UP SCREEN
900 CALL CLEAR
910 CALL HCHAR(4,5,124)
920 CALL HCHAR(19,5,125)
930 CALL HCHAR(4,28,126)
940 CALL HCHAR(19,28,127)
950 CALL VCHAR(1,16,116,24)
960 CALL VCHAR(1,17,117,24)
970 CALL HCHAR(11,1,112,32)
980 CALL HCHAR(12,1,113,32)
990 FOR I=0 TO 1
1000 FOR J=0 TO 1
1010 FOR K=0 TO 7
1020 CALL HCHAR(4+8*J+K,9+8*I,128+8*(2*I+J),8)
1030 NEXT K
1040 NEXT J
1050 NEXT I
1060 CALL HCHAR(1,16,124)
1070 CALL HCHAR(2,16,125)
1080 CALL HCHAR(1,17,126)
1090 CALL HCHAR(2,17,127)
1100 CALL HCHAR(24,1,115,32)
1110 RETURN
1120 REM S/R TO DISPLAY STRING REPRESENTATION
1130 X=5+DIV2(STI)*16
1140 Y=1+MOD2(STI)*20
1150 S$=ST$(STI)
1160 FOR I=0 TO 1
1170 FOR J=0 TO 7
1180 CALL HCHAR(Y+I,X+J,ASC(SEG$(S$,8*I+J+1,1)))
1190 NEXT J
1200 NEXT I
1220 RETURN
1230 REM ASK USER FOR INITIAL VALUES
1240 MSG$(0)="TOP LEFT "
1250 MSG$(1)="BOTTOM LEFT "
1260 MSG$(2)="TOP RIGHT "
1270 MSG$(3)="BOTTOM RIGHT "
1280 FOR STI=0 TO 3
1290 CALL CLEAR
1300 PRINT "--- CHARACTER GENERATOR ---"
1310 PRINT
1320 PRINT " ";CHR$(114);" A.E.GRAY 1982"
1330 PRINT
1340 PRINT
1350 PRINT
1360 PRINT "DO YOU WANT TO SET AN"
1370 PRINT
1380 PRINT "INITIAL PATTERN IN THE"
1390 PRINT
1400 PRINT MSG$(STI);"BOX (Y/N)?"
    
```


TI-99/4A PROGRAMMING

```

1410 CALL KEY(0,KEY,STATUS)
1420 IF STATUS=0 THEN 1410
1430 IF (KEY<>78)*(KEY<>110) THEN 1460
1440 ST$(STI)="0000000000000000"
1450 GOTO 1530
1460 PRINT
1470 PRINT "PLEASE INPUT STRING."
1480 PRINT
1490 INPUT ST$(STI)
1500 GOSUB 2760
1510 IF OK THEN 1530
1520 GOT#D 1330
1530 NEXT STI
1540 RETURN
1550 REM S/R TO FIND ABS POSITION OF CURSOR
1560 X=9+DIV2(STI)*8+CURX
1570 Y=4+MOD2(STI)*8+CURY
1580 RETURN
1590 REM POSITION CURSOR
1600 GOSUB 1550
1610 CALL GCHAR(Y,X,CH)
1620 IF CH=120 THEN 1650
1630 CALL HCHAR(Y,X,121)
1640 GOTO 1660
1650 CALL HCHAR(Y,X,122)
1660 RETURN
1670 REM REMOVE CURSOR
1680 GOSUB 1550
1690 CALL GCHAR(Y,X,CH)
1700 IF CH=122 THEN 1730
1710 CALL HCHAR(Y,X,128+8*STI)
1720 GOTO 1740
1730 CALL HCHAR(Y,X,120)
1740 RETURN
1750 REM MOVE UP
1760 IF CURY=0 THEN 1790
1770 CURY=CURY-1
1780 RETURN
1790 IF MOD2(STI)=0 THEN 1840
1800 GOSUB 2920
1810 CURY=7
1820 STI=STI-1
1830 RETURN
1840 GOSUB 2190
1850 RETURN
1860 REM MOVE DOWN
1870 IF CURY=7 THEN 1900
1880 CURY=CURY+1
1890 RETURN
1900 IF MOD2(STI)=1 THEN 1950
1910 GOSUB 2920
1920 CURY=0
1930 STI=STI+1
1940 RETURN
1950 GOSUB 2190
1960 RETURN
1970 REM MOVE RIGHT
1980 IF CURX=7 THEN 2010
1990 CURX=CURX+1
2000 RETURN
2010 IF DIV2(STI)=1 THEN 2060
2020 GOSUB 2920
2030 CURX=0
2040 STI=STI+2
2050 RETURN
2060 GOSUB 2190
2070 RETURN
2080 REM MOVE LEFT
2090 IF CURX=0 THEN 2120
2100 CURX=CURX-1
2110 RETURN
2120 IF DIV2(STI)=0 THEN 2170
2130 GOSUB 2920
2140 CURX=7
2150 STI=STI-2
2160 RETURN
2170 GOSUB 2190
2180 RETURN
2190 REM MAKE SOUND
2200 CALL SOUND(100,440,2)
2210 CALL SOUND(100,550,2)
2220 CALL SOUND(100,440,2)
2230 CALL SOUND(100,660,2)
2240 CALL SOUND(100,880,2)
2250 RETURN
2260 REMS/R TO OUTPUT LARGE LETTER AND INITIALISE REP.
2270 S$=ST$(STI)
2280 IF S$<>"0000000000000000" THEN 2290 ELSE 2410
2290 X=9+DIV2(STI)*8
2300 Y=4+MOD2(STI)*8

```

```

2310 FOR DIGIT=0 TO 15
2320 GOSUB 2460
2330 FOR J=3 TO 0 STEP -1
2340 BIT=MOD2(HEXDIGIT)
2350 HEXDIGIT=DIV2(HEXDIGIT)
2360 IF BIT<>1 THEN 2380
2370 CALL HCHAR(Y+DIV2(DIGIT),X+MOD2(DIGIT)*4+J,120)
2380 NEXT J
2390 NEXT DIGIT
2400 CALL CHAR(124+STI,S$)
2410 FOR DIGIT=0 TO 15
2420 GOSUB 2460
2430 REP(STI,DIGIT)=HEXDIGIT
2440 NEXT DIGIT
2450 RETURN
2460 REM EXTRACT DIGIT th HEXDIGIT FROM S$
2470 HEXDIGIT=ASC(SEG$(S$,DIGIT+1,1))
2480 IF HEXDIGIT<65 THEN 2510
2490 HEXDIGIT=HEXDIGIT-55
2500 RETURN
2510 HEXDIGIT=HEXDIGIT-48
2520 RETURN
2530 REM CHANGE PART OF CHAR
2540 CHANGED=-1
2550 GOSUB 1550
2560 CALL GCHAR(Y,X,CH)
2570 IF CH=120 THEN 2610
2580 NEWBIT=1
2590 CALL HCHAR(Y,X,120)
2600 GOTO 2630
2610 NEWBIT=-1
2620 CALL HCHAR(Y,X,128+8*STI)
2630 REM
2640 BITPOS=8*CURY+CURX
2650 DIGIT=DIV4(BITPOS)
2660 BITPOS=MOD4(BITPOS)
2670 ON BITPOS+1 GOTO 2740,2720,2700,2680
2680 REP(STI,DIGIT)=REP(STI,DIGIT)+NEWBIT
2690 RETURN
2700 REP(STI,DIGIT)=REP(STI,DIGIT)+2*NEWBIT
2710 RETURN
2720 REP(STI,DIGIT)=REP(STI,DIGIT)+4*NEWBIT
2730 RETURN
2740 REP(STI,DIGIT)=REP(STI,DIGIT)+8*NEWBIT
2750 RETURN
2760 REM S/R TO CHECK INPUT STRING
2770 OK=0
2780 S$=ST$(STI)
2790 IF LEN(S$)=16 THEN 2830
2800 PRINT
2810 PRINT "STRING MUST BE 16 CHARACTERS"
2820 RETURN
2830 FOR I=1 TO 16
2840 CH=ASC(SEG$(S$,I,1))
2850 IF ((CH>=48)*(CH<=57))+((CH>=65)*(CH<=70))
    THEN 2890
2860 PRINT
2870 PRINT "STRING HAS NON-HEX DIGIT."
2880 RETURN
2890 NEXT I
2900 OK=-1
2910 RETURN
2920 REM UPDATE DISPLAY WITH MODIFIED CHARACTER
2930 IF CHANGED THEN 2950
2940 RETURN
2950 CHANGED=0
2960 GOSUB 2190
2970 S$=""
2980 X=9+8*DIV2(STI)
2990 Y=4+8*MOD2(STI)
3000 FOR J=0 TO 15
3010 HEXDIGIT=REP(STI,J)
3020 IF HEXDIGIT>9 THEN 3050
3030 S$=S$&CHR$(HEXDIGIT+48)
3040 GOTO 3060
3050 S$=S$&CHR$(HEXDIGIT+55)
3060 NEXT J
3070 ST$(STI)=S$
3080 CALL CHAR(124+STI,S$)
3090 GOSUB 1120
3100 GOSUB 2190
3110 RETURN

```



Learning how to make faces

Remember how long it took you to learn the keyboard? David Gray's TI 99/4A program teaches youngsters where the keys are and rewards correct responses

Faces is a very simple game to acquaint young children with the use of the keyboard.

The program simply displays large versions of the letters A to Z and the child must press the key for the displayed letter. Pressing the wrong key results in a beep; pressing the right key is rewarded by a face at the top of the screen.

When all the letters have been dealt with, pressing the space bar causes the entire game to be repeated.

How it works

To display a face or a large version of a letter the program uses four ordinary TI 99/4A characters arranged as shown in Figure 1. The patterns needed for these sets of four characters are supplied in data statements (Lines 700 to 1260).

Line-by-line details of how the program works:

140-180 clear the screen and set the colours to be used in the program. Line 170 sets the colours of the letters and line 180 sets the colours of the faces.

190-230 read in the four patterns which define a face and assign them to the characters 152 to 155.

250 sets the variable LETTER to 1 to indicate the first letter (A).

260-550 make up the main loop of the program. This loop works by having two nested FOR statements which enumerate all the positions at which a face may be placed.

We terminate this loop when there are no more letters, ie we do not wait until there are no more positions for faces. Instead we jump out (line 530) when we have no more letters. 280-320 read in the four patterns which define the next letter and assign the patterns to characters 128-131.

330-370 display a large version of the next letter in the centre of the screen.

380-440 wait for the user to type in the correct letter; either in upper (line 410) or lower (line 420) case. Any invalid input results in a beep.

450-490, once valid input has been received, display a face at the current (X,Y) position.

500-530 increment LETTER. If all the letters have been processed then line 530 causes an exit from the main loop.

560-600 remove the last letter from the screen by overwriting it with spaces.

610-660 wait for the space bar to be pressed.

670-690 restore the data and jump back to the start of the program.

CHARACTER 1

CHARACTER 3



700-720: data for a face.

730-1260: data for the letters.

Hints on conversion

This program can easily be converted to run on any machine which allows user defined characters. To convert it for another machine will require

changing all the CALL statements and the DATA statements.

On the TI 99/4A computer the CALL statement is used to perform machine dependent activities by calling built-in subroutines. In this program the following subroutines are used.

CALL CLEAR
CALL SCREEN (N)

Clears the screen.
Defines the background colour of the screen.

CALL COLOR (N,M,P)

Defines background colour (M) and foreground colour (P) of the group of characters (N). In this program we define the colours of characters 152-159 in line 180 and the colours of characters 128-135 in line 170.

CALL CHAR (N,SS)

Defines a pattern (given by SS) for character code N.

CALL HCHAR (Y,X,CH)

Outputs the character with character code CH at the position (X,Y). The top left hand corner of the screen is position (1,1).

CALL KEY
(O,CH,STATUS)

Tests the keyboard to see if a key has been pressed. If a key has been pressed then a non-zero value is returned in STATUS and the character code of the pressed key is placed in CH. If no key has been pressed, zero is returned in STATUS.

CALL SOUND (D,P,V)

Generates a sound of duration D, pitch P and volume V.

TI 99/4A PROGRAM

The DATA statements supply data in the form of strings suitable for the subroutine CHAR. Converting this program

requires that this data be changed to a form suitable for the other machine.

Within this program char-

acters from the keyboard are held as integers, ie as their ASCII character codes. A conversion to another machine may require

that these be held as single character strings.

```

100 REM  FACES
110 REM      A.E.Gray
120 REM
130 REM
140 REM  Clear screen and set up colours.
150 CALL CLEAR
160 CALL SCREEN(1)
170 CALL COLOR(16,7,1)
180 CALL COLOR(13,5,1)
190 REM  Read in the characters to
      define a face
200 FOR I=1 TO 4
210 READ S$
220 CALL CHAR(151+I,S$)
230 NEXT I
240 REM  Enter the main loop of the program.
250 LETTER=1
260 FOR Y=1 TO 22 STEP 3
270 FOR X=4 TO 28 STEP 3
280 REM  Read in characters to define
      the next letter.
290 FOR I=1 TO 4
300 READ S$
310 CALL CHAR(127+I,S$)
320 NEXT I
330 REM  Display the next letter.
340 CALL HCHAR(15,14,128)
350 CALL HCHAR(16,14,129)
360 CALL HCHAR(15,15,130)
case.
390 CALL KEY(0,CH,STATUS)
400 IF STATUS=0 THEN 390
410 IF (CH-64)=LETTER THEN 460
420 IF (CH-96)=LETTER THEN 460
430 CALL SOUND(-100,220,0)
440 GOTO 390
450 REM  Display a face for getting
      the correct answer.
460 CALL HCHAR(Y,X,152)
470 CALL HCHAR(Y+1,X,153)
480 CALL HCHAR(Y,X+1,154)
490 CALL HCHAR(Y+1,X+1,155)
500 REM  Move onto the next letter.
510 LETTER=LETTER+1
520 REM  If we have finished all the letters
      then exit the main loop.
530 IF LETTER>26 THEN 560
540 NEXT X
550 NEXT Y
560 REM  Remove last letter form the screen.
570 CALL HCHAR(15,14,32)
580 CALL HCHAR(16,14,32)
590 CALL HCHAR(15,15,32)
600 CALL HCHAR(16,15,32)
610 REM  Wait for a space to be input.
620 CALL KEY(0,CH,STATUS)
630 IF STATUS=0 THEN 620
640 IF CH=32 THEN 680
650 CALL SOUND(100,440,0)
660 GOTO 620
670 REM  Start the whole
      program again.

```

```

370 CALL HCHAR(16,15,131)
380 REM  Wait for the user to input the
      correct letter ; either UPPER or lower
680 RESTORE
690 GOTO 100
700 REM  Data for a FACE.
710 DATA 0F1020448E848080,908887804020100F
720 DATA F008042271210101,0911E10102040BF0
730 REM  Letters A to Z; 4 characters each.
740 DATA 000007070C0C1B1B,1F1F1B1B1B1B
750 DATA 0000E0E030301B1B,F8F81B1B1B1B
760 DATA 0000F0F0C0C0C0F,0F0C0C0C0F0F
770 DATA 000080C0606060C0,C0606060C080
780 DATA 000003070C0C0C0C,0C0C0C0C0703
790 DATA 000080C06060
800 DATA 00006060C080
810 DATA 0000F0F0C0C0C0C,0C0C0C0C0F0F
820 DATA 000080C060606060,60606060C080
830 DATA 0000F0F0C0C0C0C,0F0C0C0C0F0F
840 DATA 0000E0E000000080,80000000E0E0
850 DATA 0000F0F0C0C0C0C,0F0C0C0C0C0C
860 DATA 0000E0E00000008,B
870 DATA 000003070C0C0C0C,0C0C0C0C0703
880 DATA 000080C06060,00F0F060E0E
890 DATA 0000C0C0C0C0C0C,0F0C0C0C0C0C
900 DATA 00003030303030F0,F0303030303
910 DATA 0000F0F01010101,010101010F0F
920 DATA 0000F0F080808080,80808080F0F
930 DATA 0000F0F01010101,010101010F07
940 DATA 0000F0F08080808,808080808
950 DATA 0000C0C0C0C0C0D0F,0F0D0C0C0C0C
960 DATA 0000303060C08,0080C060303
970 DATA 0000C0C0C0C0C0C,0C0C0C0C0F0F
980 DATA 0,00000000F0F
990 DATA 00001B1B1C1E1A1B,191B1B1B1B1B
1000 DATA 00001B1B3B7B8BD8,9B1B1B1B1B1B
1010 DATA 00001C1C1E1B1B19,191B1B1B1B1B
1020 DATA 00001B1B1B1B1B9B,9BDBD87B3B3B
1030 DATA 000003070C0C0C0C,0C0C0C0C0703
1040 DATA 0000C0E030303030,30303030E0C
1050 DATA 0000F0F0C0C0C0F,0F0C0C0C0C0C
1060 DATA 000080C0606060C,B
1070 DATA 000003070C0C0C0C,0C0C0C0C0703
1080 DATA 0000C0E030303030,3030B070F0F81B
1090 DATA 0000F0F0C0C0C0F,0F0D0C0C0C0C
1100 DATA 000080C0606060C,8080C060303
1110 DATA 000003070C0C0C0F,07000C0C0703
1120 DATA 0000C0E0303000E,F0303030E0C
1130 DATA 00001F1F01010101,010101010101
1140 DATA 0000F8F880808080,80808080808
1150 DATA 0000C0C0C0C0C0C,0C0C0C0C0707
1160 DATA 0000303030303030,30303030E0E
1170 DATA 00001B1B1B0C0C0C,060606030301
1180 DATA 00001B1B1B30303,606060C0C08
1190 DATA 00001B1B1B1B1B19,1B1A1E1C1B1B
1200 DATA 00001B1B1B1B1B9B,DB5B7B3B1B1B
1210 DATA 00002030180C0603,0103060C1B3
1220 DATA 0000040C1B3060C0,C06030180C04
1230 DATA 00001B1B1B0C0603,010101010101
1240 DATA 00001B1B1B3060C0,80808080808
1250 DATA 00001F1F00000001,03060C1B1F1F
1260 DATA 0000F8F83060C080,00000000F8F8

```



Ever fancied being a knight in shining armour? Here's your chance as you strive to rescue the maiden in distress from the black magician's castle.

Move your knight using the four arrow keys (ESDX) with the alpha-lock button down. One problem — the evil magician is throwing large boulders at you. If you are hit by, or bump into one, you fail.

A game of strategy and quick reactions, this game, although written for the Texas TI-99/4A, should be easily translatable for any computer with user-defined characters.

The difficulty of the game can be increased by changing the loop count in line 260.

How it works

160	clears screen
170	set screen colour to cyan
190-240	user defined characters
250-330	set up initial screen
340-370	initialise variables
400	look at keyboard
410-420	swap old and new positions
430-510	find direction
520	look at screen in new position
530-550	test for special situations
580-600	move knight
610-620	swap updated positions
640-750	throw rock subroutine

Up you get... and into battle with the black magician. A scene from Jabberwocky

Make a knight of it, there's a maiden in distress

Now's your chance to be a hero. All you have to do is rescue a maiden from the clutches of the evil black magician. Vince Apps' program is for the TI-99/4A

760-790	main program loop	910-1000	crashed into rock
810-840	play music	1020-1120	hit by rock
850-880	print success message	1140-1180	simulation of print @ routine

Hints on conversion

Texas Basic has some unusual commands for graphics and sound:

CALL CHAR (number, hexadecimal string) — Controls the user definition of characters. Number refers to the ASCII code of the re-defined character.

CALL HCHAR (row, column, number) — This prints the character whose ASCII code is number at position row, column on the screen. The TI99/4A has a screen size of 24 rows by 32 columns.

CALL GCHAR (row, column, variable) — This is the equivalent of PEEK. The ASCII code of the character at position row, column on the screen is left in the designated variable.

CALL SOUND (d, f, v) — Sounds a note of frequency f, duration d milliseconds and volume v.

CALL KEY (O, K, S) — TI99 equivalent of INKEY\$. Call key will leave the ASCII code of the key pressed in the variable K.

CALL CLEAR — Clears the screen. A direct equivalent of CLS.



TI-99/4A PROGRAM

```

100 REM
110 REM RESCUE
120 REM
130 REM
140 REM V.H. APPS
150 REM
160 CALL CLEAR
170 CALL SCREEN(8)
180 RANDOMIZE
190 REM DEFINE CHARACTERS
200 CALL CHAR(120,"AFFFFFB6FFFFB6FE")
210 CALL CHAR(121,"AAFEFEDAFEDA7E7E")
220 CALL CHAR(128,"08081C2A2A081414")
230 CALL CHAR(144,"387EFFFFFFFF7E")
240 CALL COLOR(15,13,1)
250 REM SET UP SCREEN
260 FOR J=1 TO 180
270 R1=INT(RND*30+3)
280 R2=INT(RND*23+2)
290 CALL HCHAR(R2,R1,144)
300 NEXT J
310 CALL HCHAR(1,30,120)
320 CALL HCHAR(1,31,121)
330 CALL HCHAR(24,5,128)
340 MRN=24
350 MRO=24
360 MCO=5
370 MCN=5
380 GOTO 770
390 REM MOVE MAN
400 CALL KEY(0,K,S)
410 MRN=MRO
420 MCN=MCO
430 IF K<>83 THEN 450
440 MCN=MCN-1
450 IF K<>68 THEN 470
460 MCN=MCN+1
470 IF K<>69 THEN 490
480 MRN=MRN-1
490 IF K<>88 THEN 510
500 MRN=MRN+1
510 IF (MCN<3)+(MCN>32)+(MRN<1)+(MRN>24) THEN 630
520 CALL GCHAR(MRN,MCN,X)
530 IF X=32 THEN 570
540 IF (X=120)+(X=121) THEN 810
550 IF X=144 THEN 910
560 GOTO 630
570 IF (MRN=MRO)*(MCN=MCO) THEN 630
580 CALL HCHAR(MRO,MCO,32)
590 CALL SOUND(100,330,2)
600 CALL HCHAR(MRN,MCN,128)
610 MRO=MRN
620 MCO=MCN
630 RETURN
640 REM THROW ROCK
650 R1=INT(RND*12)+MRO-6
660 IF (R1>24)+(R1<1) THEN 650
670 R2=INT(RND*12)+MCO-6
680 IF (R2>32)+(R2<3) THEN 670
690 CALL GCHAR(R1,R2,X)
700 IF X=32 THEN 730
710 IF X=120 THEN 1030
720 IF (X=120)+(X=121) THEN 750
730 CALL SOUND(50,160,3)
740 CALL HCHAR(R1,R2,144)
750 RETURN
760 REM MAIN LOOP
770 GOSUB 400
780 GOSUB 650
790 GOTO 770
800 REM SUCCESS
810 CALL SOUND(500,330,2)
820 CALL SOUND(500,450,2)
830 CALL SOUND(500,360,2)
840 CALL SOUND(900,500,2)
850 M$="YOU MADE IT !!"
860 X=10
870 GOSUB 1150
880 GOSUB 1210
890 INPUT "PLAY AGAIN? ":Q$
900 IF SEG$(Q$,1,1)="N" THEN 1130 ELSE 160

```

```

910 FOR J=1 TO 10
920 CALL HCHAR(MRO,MCO,32)
930 CALL SOUND(100,-5,2)
940 CALL HCHAR(MRO,MCO,128)
950 NEXT J
960 REM WRONG TURNING
970 M$="YOU CRASHED !!"
980 X=10
990 GOSUB 1150
1000 GOSUB 1210
1010 GOTO 890
1020 REM ROCK HIT
1030 M$="S P L A T !!"
1040 X=11
1050 FOR J=1 TO 10
1060 CALL HCHAR(MRO,MCO,32)
1070 CALL HCHAR(MRO,MCO,144)
1080 CALL SOUND(100,220,2)
1090 NEXT J
1100 GOSUB 1150
1110 GOSUB 1210
1120 GOTO 890
1130 END
1140 REM PRINT AT
1150 FOR J=1 TO LEN(M$)
1160 CH=ASC(SEG$(M$,J,1))
1170 CALL HCHAR(11,X+J,CH)
1180 NEXT J
1190 RETURN
1200 REM DELAY
1210 FOR D=1 TO 200
1220 NEXT D
1230 RETURN

```



Fast reactions are essential for a knight. Nigel Terry and Nicholas Clay in *Excalibur*

Bank on your micro to check on your cheques

How much is in your bank account? Don't wait for a statement, Anthony Gray's program for the TI-99/4A will tell you straight away, and help your budgetting

Keep an eye on your spending with my bank account program for the TI-99/4A. It allows the initial balance and a series of transactions to be maintained and manipulated.

Each entry consists of an amount, a date, a cheque number and some details.

The amount is entered as the number of pence — a negative number is a debit and a positive number represents a credit. Dates are entered in the form dd/mm/yy and are checked for validity.

The cheque numbers and details are arbitrary strings of up to 15 and 56 characters respectively: extra characters are discarded. This means two transactions can always fit into a record on a cassette tape.

When the program is run you are asked for the data and the following menu of options is displayed, chosen by pressing the appropriate letter:

- N** set New date and/or new initial balance
- B** determine the new Balance after transactions
- I** Input details of an account from a cassette file
- O** Output. Save details of the account on a cassette file

- A** Add a new entry to the account
- C** Change an existing entry
- R** Remove an entry from the account
- S** Sort all the entries according to their dates
- T** Terminate. Quit the program

You should only need to follow the instructions on the screen, but bear in mind the following points:

An entry is specified by its item number (1, 2, 3...)

When you want to change an entry, that entry will be displayed along with another menu. Press the appropriate letter for the option you require

When asked a question with a yes or no answer press Y or y for yes and N or n for no

The program has been designed to be as robust as possible, however, there are certain errors with tapes which cannot be detected and prevented, so take care.

No attempt is made to ensure that only integer values are input as amounts or that values do not become too large. Numeric errors of this kind do not cause the program to fail, but will give wrong results.

How it works

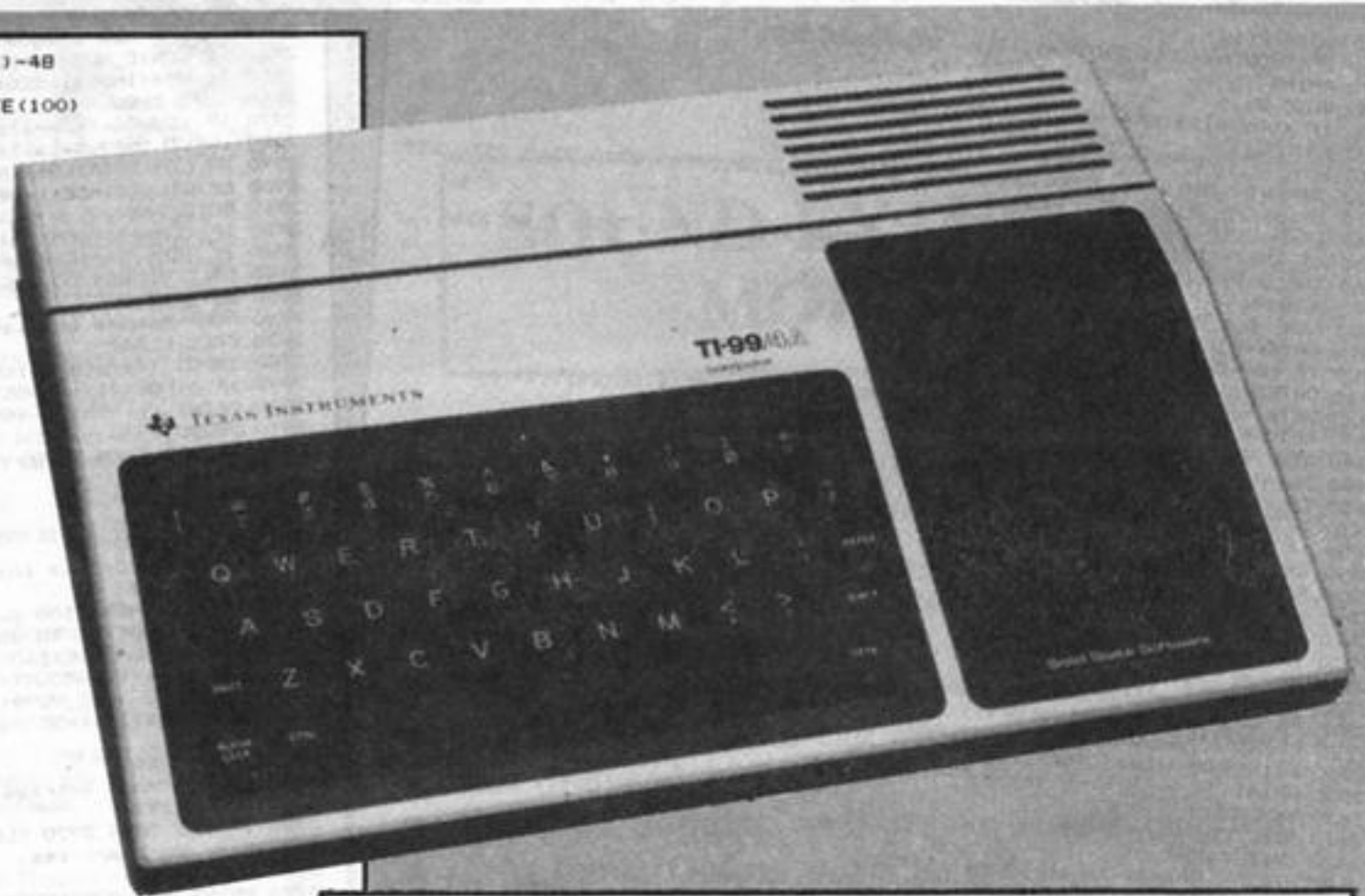
- 100 function to convert two-character string into a number
- 110 maximum number of entries possible
- 120 arrays to hold details, amounts, cheque numbers and dates of entries
- 130 definition of £ character
- 140-160 set initial values for the number of entries (COUNT), today's date and balance
- 170 get today's date
- 180 display menu
- 190-320 await input and act accordingly. Accept either upper or lower case and make noise on invalid selection
- 330-370 deal with Terminate option, but first ensure user wants to quit
- 380-440 sub-routine to display menu
- 610-720 deal with New. Initialise option. Ask for today's date and then, if current initial balance is incorrect, get a new one
- 730-860 deal with Balance option. Work out new balance by adding AMOUNTs to existing balance
- 870-1140 Input file option. First ensure user does want to read a file then ensure title of file indicates correct data. Read data from tape and display cheque number of each entry read
- 1150-1390 Output file option. First ask user for a title for the file being written and then ensure all is well before writing the tape. As each item is written to the tape, display its cheque number
- 1400-1710 Add entry option. After checking there is room, input data for the new entry. Display new entry and ask for confirmation that it is to be inserted. Repeat if user wants to add more
- 1720-1860 Display entries option. Display each entry or message to show there are none. Waits for user to press space bar after each pair of entries has been displayed
- 1870-1960 Sub-routine to turn a number (VALUE) representing a number of pence into a string (VALUES) in sterling notation. For example, 5673 is turned into £56.73 and -5673 is turned into £56.73 db.
- 1970-2070 sub-routine to accept Y, y, N or n as answers and return true or false on yes
- 2080-2370 request date as dd/mm/yy, check validity and place date into THE DATE in the form yymmdd to ensure that entries can be sorted into chronological order
- 2380-2430 sub-routine to convert date back into dd/mm/yy and place in DATES
- 2440-2490 press space bar to continue
- 2500-2700 display entry with item number given by INDEX
- 2610-2950 Change entry option. Ask for item number of entry to be changed and check it exists. Then display entry and menu of options. Wait for selection and perform required change. By selecting Next, user can change another entry and by choosing Exit will return to main menu
- 2960-3190 Remove entry. Ask for item number to be removed, ensure it exists, display entry, ask for confirmation that it has to be removed. Move all subsequent entries down one place. Check if user wants to remove more, return to main menu
- 3200-3400 Sort entries into chronological order

TI-99/4A PROGRAM

```

100 DEF CONV(S%)=10*(ASC(S%)-48)+ASC(SEG$(S%,2,1))-48
110 DEF MAX=100
120 DIM DETAILS$(100),AMOUNT(100),C_NUM$(100),DATE(100)
130 CALL CHAR(128,"16127E107C305C73")
140 COUNT=0
150 TODAY=830101
160 BALANCE=0
170 GOSUB 380
180 GOSUB 450
190 CALL KEY(0,CH,STATUS)
200 IF STATUS=0 THEN 190
210 IF (CH=78)+(CH=110) THEN 610
220 IF (CH=66)+(CH=98) THEN 730
230 IF (CH=73)+(CH=105) THEN 870
240 IF (CH=79)+(CH=111) THEN 1150
250 IF (CH=65)+(CH=97) THEN 1400
260 IF (CH=67)+(CH=99) THEN 2610
270 IF (CH=82)+(CH=114) THEN 2960
280 IF (CH=68)+(CH=100) THEN 1720
290 IF (CH=83)+(CH=115) THEN 3200
300 IF (CH=84)+(CH=116) THEN 330
310 CALL SOUND(200,220,0)
320 GOTO 190
330 CALL CLEAR
340 PRINT "Are you sure you want to ""quit?""
350 GOSUB 1970
360 IF YES THEN 370 ELSE 180
370 STOP
380 REM Get to-day's date.
390 CALL CLEAR
400 PRINT "BANK BALANCE PROGRAM""
410 PRINT "Please input to-day's date.""
420 GOSUB 2080
430 TODAY=THE_DATE
440 RETURN
450 REM Display menu.
460 CALL CLEAR
470 THE_DATE=TODAY
480 GOSUB 2380
490 PRINT "Date is ""DATE""
500 PRINT TAB(8);"New, Initialise.""
510 PRINT TAB(8);"B(alance).""
520 PRINT TAB(8);"I(nput file).""
530 PRINT TAB(8);"O(utput file).""
540 PRINT TAB(8);"A(dd entry).""
550 PRINT TAB(8);"C(hange entry).""
560 PRINT TAB(8);"R(emove entry).""
570 PRINT TAB(8);"D(isplay entries).""
580 PRINT TAB(8);"S(ort entries).""
590 PRINT TAB(8);"T(erminate).""
600 RETURN
610 REM Get new date and balance.
620 GOSUB 380
630 CALL CLEAR
640 VALUE=BALANCE
650 GOSUB 1870
660 PRINT "Is the initial balance :- ""TAB(6);VALUE"" ?"
670 GOSUB 1970
680 CALL CLEAR
690 IF YES THEN 180
700 PRINT "What is the initial balance""in pence.""
710 INPUT BALANCE
720 GOTO 180
730 REM Work out new balance.
740 CALL CLEAR
750 TEMP=BALANCE
760 FOR I=1 TO COUNT
770 TEMP=TEMP+AMOUNT(I)
780 NEXT I
790 VALUE=BALANCE
800 GOSUB 1870
810 PRINT "Old balance :-"" ""VALUE""
820 VALUE=TEMP
830 GOSUB 1870
840 PRINT "New balance :-"" ""VALUE"" ""after""COUNT""transactions.""
850 GOSUB 2440
860 GOTO 180
870 REM Input details from tape.
880 CALL CLEAR
890 PRINT "Are you sure you want to""read a tape?""
900 GOSUB 1970
910 IF YES THEN 920 ELSE 180
920 CALL CLEAR
930 OPEN #1:"CSI",INTERNAL,INPUT,FIXED 192
940 INPUT #1:THE_DATE,TITLE$,
950 CALL CLEAR
960 GOSUB 2380
970 PRINT "TAPE IS :-""
980 PRINT " ""TITLE$
990 PRINT ""Written on ""DATE$""
1000 PRINT "Do you want to read it?""
1010 GOSUB 1970
1020 IF YES THEN 1030 ELSE 1120
1030 CALL CLEAR
1040 INPUT #1:COUNT,BALANCE
1050 FOR INDEX=1 TO COUNT
1060 IF (INDEX-2*INT(INDEX/2))=0 THEN 1090
1070 INPUT #1:DETAILS$(INDEX),AMOUNT(INDEX),C_NUM$(INDEX),DATE(INDEX),
1080 GOTO 1100
1090 INPUT #1:DETAILS$(INDEX),AMOUNT(INDEX),C_NUM$(INDEX),DATE(INDEX)
1100 PRINT INDEX;" -- ""C_NUM$(INDEX)""
1110 NEXT INDEX
1120 CLOSE #1
1130 CALL CLEAR
1140 GOTO 180
1150 REM Save details on tape.
1160 CALL CLEAR
1170 PRINT "Title for tape please.""
1180 INPUT TITLE$
1190 IF LEN(TITLE$)<100 THEN 1210
1200 TITLE$=SEG$(TITLE$,1,99)
1210 CALL CLEAR
1220 PRINT "Are you sure you want to ""write the tape :-""TITLE$
1230 GOSUB 1970
1240 IF YES THEN 1250 ELSE 180
1250 CALL CLEAR
1260 PRINT "Writing :-""TITLE$
1270 OPEN #1:"CSI",INTERNAL,OUTPUT,FIXED 192
1280 PRINT #1:TODAY,TITLE$,
1290 PRINT #1:COUNT,BALANCE
1300 FOR INDEX=1 TO COUNT
1310 PRINT INDEX;" -- ""C_NUM$(INDEX)""
1320 IF (INDEX-2*INT(INDEX/2))=0 THEN 1350
1330 PRINT #1:DETAILS$(INDEX),AMOUNT(INDEX),C_NUM$(INDEX),DATE(INDEX),
1340 GOTO 1360
1350 PRINT #1:DETAILS$(INDEX),AMOUNT(INDEX),C_NUM$(INDEX),DATE(INDEX)
1360 NEXT INDEX
1370 CLOSE #1
1380 CALL CLEAR

```



```

1390 GOTO 180
1400 REM Add a new entry.
1410 CALL CLEAR
1420 IF COUNT<MAX THEN 1460
1430 PRINT "No room for more entries""
1440 GOSUB 2440
1450 GOTO 180
1460 COUNT=COUNT+1
1470 PRINT "Input date for the entry.""
1480 GOSUB 2080
1490 DATE(COUNT)=THE_DATE
1500 PRINT ""
1510 INPUT "Amount in pence : ""AMOUNT(COUNT)
1520 PRINT ""
1530 INPUT "Cheque number : ""C_NUM$(COUNT)
1540 IF LEN(C_NUM$(COUNT))>15 THEN 1550 ELSE 1560
1550 C_NUM$(COUNT)=SEG$(C_NUM$(COUNT),1,15)
1560 PRINT ""
1570 INPUT "Details : ""DETAILS$(COUNT)
1580 IF LEN(DETAILS$(COUNT))>56 THEN 1590 ELSE 1600
1590 DETAILS$(COUNT)=SEG$(DETAILS$(COUNT),1,56)
1600 CALL CLEAR
1610 INDEX=COUNT
1620 GOSUB 2500
1630 PRINT "Do you want this inserted?""
1640 GOSUB 1970
1650 IF YES THEN 1670
1660 COUNT=COUNT+1
1670 CALL CLEAR
1680 PRINT "Do you want to add another""entry?""
1690 GOSUB 1970
1700 IF YES THEN 1400
1710 GOTO 180
1720 REM Display entries.
1730 CALL CLEAR
1740 IF COUNT<>0 THEN 1780
1750 PRINT "There are no entries.""
1760 GOSUB 2440
1770 GOTO 180
1780 FOR INDEX=1 TO COUNT
1790 GOSUB 2500
1800 IF (INDEX-2*INT(INDEX/2))<>0 THEN 1830
1810 GOSUB 2440
1820 CALL CLEAR
1830 NEXT INDEX
1840 PRINT " --- FINISHED ---""
1850 GOSUB 2440
1860 GOTO 180
1870 REM Turn VALUE into a suitable string VALUE$.
1880 NEG=VALUE<0
1890 VALUE=ABS(VALUE)
1900 VALUE$=STR$(VALUE)
1910 IF LEN(VALUE$)>2 THEN 1930
1920 VALUE$=SEG$("00",1,3-LEN(VALUE$))+VALUE$
1930 VALUE$=CHR$(128)+" "+SEG$(VALUE$,1,LEN(VALUE$)-2)+"
"+SEG$(VALUE$,LEN(VALUE$)-1,2)
1940 IF NEG THEN 1950 ELSE 1960
1950 VALUE$=VALUE$+" db"
1960 RETURN
1970 REM Get answer Y or N.
1980 CALL KEY(0,CH,STATUS)
1990 IF STATUS=0 THEN 1980
2000 IF (CH=89)+(CH=121) THEN 2010 ELSE 2030
2010 YES=-1
2020 RETURN
2030 IF (CH=78)+(CH=110) THEN 2040 ELSE 2060
2040 YES=0
2050 RETURN
2060 CALL SOUND(200,220,0)
2070 GOTO 1980
2080 REM Get a date.
2090 THE_DATE=TODAY
2100 GOSUB 2380
2110 PRINT "Enter date as dd/mm/yy."" ENTER gives ""DATE""
2120 INPUT D$
2130 IF D$="" THEN 2140 ELSE 2150
2140 RETURN
2150 X=POS(D$,"/",1)
2160 Y=POS(D$,"/",X+1)
2170 IF (X=3)+(Y=6)+(LEN(D$)=8) THEN 2200
2180 PRINT "" INVALID FORMAT.""
2190 GOTO 2110
2200 MONTH=CONV(SEG$(D$,4,2))
2210 YEAR=CONV(SEG$(D$,7,2))
2220 DAY=CONV(SEG$(D$,1,2))
2230 IF (YEAR=83)+(YEAR<99) THEN 2260 ELSE 2240
2240 PRINT "" INVALID YEAR.""

```


TI-99/4A PROGRAM

```

2250 GOTO 2110
2260 IF (MONTH=1) THEN 2290 ELSE 2270
2270 PRINT "INVALID MONTH.":
2280 GOTO 2110
2290 IF (DAY=1) THEN 2320 ELSE 2300
2300 PRINT "INVALID DAY.":
2310 GOTO 2110
2320 ON MONTH GOTO 2340,2350,2340,2330,2340,2340,2330,2340,2340,2330,2340,2340
2330 IF DAY>30 THEN 2300 ELSE 2360
2340 IF DAY>31 THEN 2300 ELSE 2360
2350 IF (DAY=29) THEN ((YEAR=4) AND (YEAR/4)<>0)) THEN 2300 ELSE 2360
2360 THE_DATE=10000*YEAR+100*MONTH+DAY
2370 RETURN
2380 REM Put THE_DATE into DATE$.
2390 DATE$=STR$(THE_DATE)
2400 IF LEN(DATE$)=6 THEN 2420
2410 DATE$="0"&DATE$
2420 DATE$=SEG$(DATE$,5,2)&"/"&SEG$(DATE$,3,2)&"/"&SEG$(DATE$,1,2)
2430 RETURN
2440 REM Space to continue.
2450 PRINT "PRESS SPACE TO CONTINUE."
2460 CALL KEY(0,CH,STATUS)
2470 IF STATUS=0 THEN 2460
2480 IF CH<>" " THEN 2460
2490 RETURN
2500 REM Display entry INDEX.
2510 VALUE=AMOUNT(INDEX)
2520 GOSUB 1870
2530 THE_DATE=DATE(INDEX)
2540 GOSUB 2380
2550 PRINT "Item : ";INDEX;" Date : ";THE_DATE:
2560 PRINT "Amount : ";VALUE:
2570 PRINT "Cheque : ";C_NUM$(INDEX):
2580 PRINT DETAILS$(INDEX):
2590 PRINT "-----"
2600 RETURN
2610 REM Change entries.
2620 CALL CLEAR
2630 INPUT "Change which entry : ";INDEX
2640 IF (INDEX<1) OR (INDEX>COUNT) THEN 2650 ELSE 2680
2650 PRINT "This entry does not exist.":
2660 GOSUB 1970
2670 IF YES THEN 2620 ELSE 180
2680 CALL CLEAR
2690 GOSUB 2500
2700 PRINT "Date Amount Cheque":
2710 CALL KEY(0,CH,STATUS)
2720 IF STATUS=0 THEN 2710
2730 IF (CH=65) OR (CH=97) THEN 2740 ELSE 2790
2740 CALL CLEAR
2750 PRINT "Please input new date.":
2760 GOSUB 2080
2770 DATE(INDEX)=THE_DATE
2780 GOTO 2680
2790 IF (CH=65) OR (CH=97) THEN 2800 ELSE 2820
2800 INPUT "New amount in pence : ";AMOUNT(INDEX)
2810 GOTO 2680
2820 IF (CH=67) OR (CH=99) THEN 2830 ELSE 2870

```

```

2830 INPUT "New cheque number : ";C_NUM$(INDEX)
2840 IF LEN(C_NUM$(INDEX))<15 THEN 2680
2850 C_NUM$(INDEX)=SEG$(C_NUM$(INDEX),1,15)
2860 GOTO 2680
2870 IF (CH=84) OR (CH=116) THEN 2880 ELSE 2920
2880 INPUT "New details : ";DETAILS$(INDEX)
2890 IF LEN(DETAILS$(INDEX))<56 THEN 2680
2900 DETAILS$(INDEX)=SEG$(DETAILS$(INDEX),1,56)
2910 GOTO 2680
2920 IF (CH=69) OR (CH=101) THEN 180 ELSE 2930
2930 IF (CH=78) OR (CH=110) THEN 2620
2940 CALL SOUND(200,220,0)
2950 GOTO 2710
2960 REM Remove entries.
2970 CALL CLEAR
2980 INPUT "Delete which entry : ";INDEX
2990 IF (INDEX<1) OR (INDEX>COUNT) THEN 3000 ELSE 3040
3000 PRINT "This entry does not exist.":
3010 GOSUB 1970
3020 IF YES THEN 2970
3030 GOTO 180
3040 CALL CLEAR
3050 COUNT=COUNT-1
3060 GOSUB 2500
3070 PRINT "Delete this entry?":
3080 GOSUB 1970
3090 IF YES THEN 3100 ELSE 3160
3100 FOR I=INDEX+1 TO COUNT
3110 DATE(I-1)=DATE(I)
3120 AMOUNT(I-1)=AMOUNT(I)
3130 C_NUM$(I-1)=C_NUM$(I)
3140 DETAILS$(I-1)=DETAILS$(I)
3150 NEXT I
3160 CALL CLEAR
3170 PRINT "More entries to delete?":
3180 GOSUB 1970
3190 IF YES THEN 2970 ELSE 180
3200 REM Sort entries.
3210 CALL CLEAR
3220 PRINT "...SORTING":
3230 EXS=0
3240 FOR I=1 TO COUNT-1
3250 IF DATE(I)<DATE(I+1) THEN 3390
3260 TEMP=DATE(I)
3270 DATE(I)=DATE(I+1)
3280 DATE(I+1)=TEMP
3290 TEMP=AMOUNT(I)
3300 AMOUNT(I)=AMOUNT(I+1)
3310 AMOUNT(I+1)=TEMP
3320 TEMP=DETAILS$(I)
3330 DETAILS$(I)=DETAILS$(I+1)
3340 DETAILS$(I+1)=TEMP
3350 TEMP=C_NUM$(I)
3360 C_NUM$(I)=C_NUM$(I+1)
3370 C_NUM$(I+1)=TEMP
3380 EXS=EXS+1
3390 NEXT I
3400 IF EXS THEN 3230 ELSE 180

```

SIR COMPUTERS LTD — CARDIFF

Agents for Acorn, BBC and TORCH Computers

BBC Microcomputers

Model B	£399.00
BBC Model B with 1.2 OS	£399.00
Model B with disc interface	£509.00
Single 100K disc drive	£249.00
Dual 2 x 100K disc drive	£389.00
The disc manual and utilities disc are both included.	
Disc interface for the BBC Micro	
Kit	£95.00
Fitted	£110.00
Upgrade of BBC Model A to B	£90.00
Please telephone for up to date information on Prestel, Teletext, speech synthesis, second processors, etc.	

TORCH Computers

Z-80 Disc Pack for the BBC Microcomputer £895.00

This unit connects to the BBC Micro in the same way as a normal disc drive, but as well as offering a dual 2 x 400K disc drive for use under BBC BASIC or other languages it provides the option of using the wide range of CP/M software available for business and data processing applications. The firmware supplied with the machine allows switching between BASIC and CPN, a powerful operating system developed from CP/M 2.2.

In addition to the disc pack a second processor is supplied. This is a Z-80A with its own 64K RAM card, communicating with the 6502A in the BBC computer through the 'Tube'. Typically the speed of execution of programs under the twin-processor system is increased by up to 50% compared with a conventional single-processor computer. A third processor, the 16 bit 68000, will shortly be available.

TORCH CF240 £2795.00 (ex VAT)

This is an extension of the BBC microcomputer/Torch disc pack system, available in a single unit. The computer contains a BBC-based peripheral processor connected to the main Z-80 computer, a dual 2 x 400K disc drive as described above, a high resolution (80 character) colour monitor and a complete British Telecom approved 1200 baud modem. It is the only microcomputer which has been granted permission for direct connection to the Public Switched Telephone Network both in the U.K. and the United States.

The TORCH can communicate either directly with another TORCH or with virtually any other type of computer via Prestel. Using the Gateway facility of Prestel it is possible for the TORCH to access vast amounts of information stored by private organisations on public database systems. The Mailbox facility of Prestel also allows the use of electronic mail.

TORCH CH240/10 As above but with a 10 MB hard disc drive.
TORCH CH240/21 As above but with a 21 MB hard disc drive.

PERIPHERALS

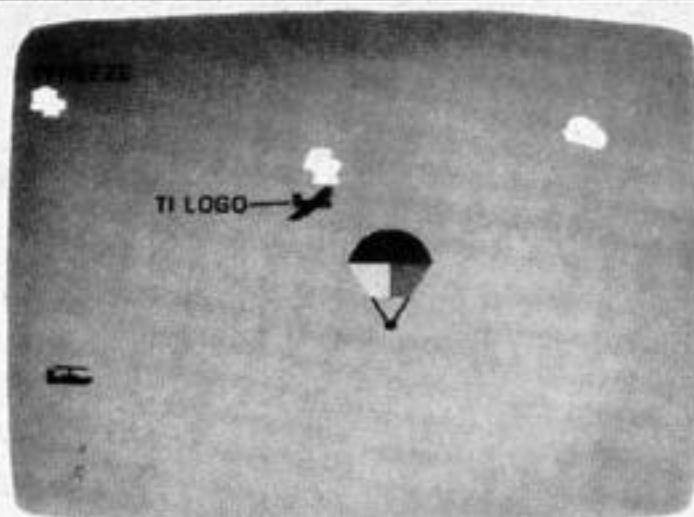
Seikosha GP 100A Printer	£229.00
Epson MX 80 F/T type 3 Printer	£399.00
NEC PC 8023 Printer	£389.00
Kaga 12" RGB Monitor	£280.00
Sanyo 14" RGB Monitor	£270.00
High resolution 12" black/green monitor	£85.00

SOFTWARE

We currently hold in stock programs from the following suppliers! Acornsoft, A&F Software, Bug Byte, Computer Concepts, Data Base Software, Digital Fantasia, Gemini, Golem, IJK Software, L'Level 9 Software, Molimerx, MP Software, Program Power, Salamander Software, Software for All, Superior Software.

Unfortunately, we are unable to supply software by mail except as part of a large order. Delivery by Interlink of any of the above items £10.00 Unless otherwise stated all prices include VAT.

SIR Computers Ltd., 91 Whitchurch Road, Cardiff. Telephone: (0222) 21341



The TI 99/4A Home Computer

This microcomputer is based on TMS9900 16 bit microprocessor. It includes 16K RAM, 16 colour high resolution graphic (192x256). The screen display is 32 characters, 24 lines TI-BASIC. Full size keyboard. For Software there are about 1,000 Programs to choose from. There are a lot of peripherals available e.g. Disk Drives, Disk Interface, Speech Synthesizer, Extra RAM. Additional Languages (PASCAL, TI-LOGO, ASSEMBLER).

TI 99/4A for just £149.95

TWO SPECIAL OFFERS

OFFERS END JULY 2nd 1983



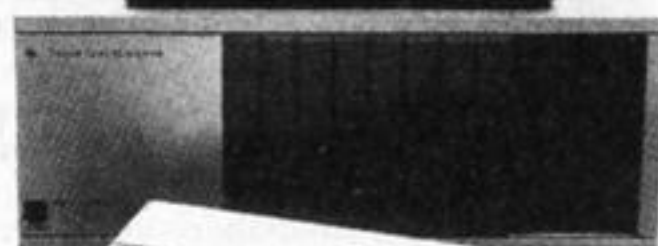
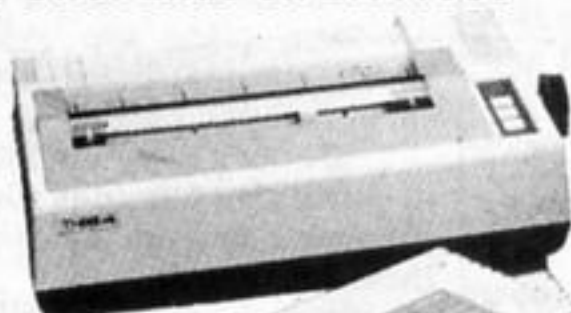
WHEN YOU BUY SIX SOFTWARE MODULES TEXAS INSTRUMENTS WILL SEND THE SOLID STATE

FREE!

SPEECH SYNTHESIZER OR THE

T.I. PROGRAM TAPE RECORDER
Hurry! This Offer Ends on 2.7.1983

PLEASE
SEND A
LARGE
S.A.E.
FOR
DETAILS



WHEN YOU BUY A TI 99/4A HOME COMPUTER TEXAS INSTRUMENTS WILL SEND YOU

FREE!

A PAIR OF JOYSTICKS,

Connect Four Games Cartridge and Beginners Basic Tutor Cassette. (approx value £50)

Hurry! This incredible offer ends on 2.7.1983

TI SOFTWARE

ENTERTAINMENT SOFTWARE	MEDIA	PRICE INC VAT
Parsec	Module	£25.95
TI-Invaders	Module	£17.95
Video Games 1	Module	£20.95
Mind Changers	Module	£12.95
A-Maze-Ing	Module	£17.95
Connect Four	Module	£17.95
Soccer	Module	£20.95
Tombstone City	Module	£20.95
Wumpus	Module	£17.95
Zero Zap	Module	£17.95
Attack	Module	£17.95
Blasto	Module	£17.95
Hustle	Module	£17.95
Yahtzee	Module	£17.95
Blackjack/Poker	Module	£17.95
Car Wars	Module	£17.95
Munchman	Module	£25.95
Chess	Module	£34.95
Adventure & Pirate	Module	£25.95
Othello	Module	£25.95
Alpine	Module	£25.95
Chisholm Trail	Module	£25.95
ADDITIONAL ADVENTURE PROGRAMS TO BE USED WITH THE ADVENTURE MODULE		
Adventureland	Cassette	£17.95
Mission Impossible	Cassette	£17.95
Voodoo Castle	Cassette	£17.95
The Count	Cassette	£17.95
Strange Odyssey	Cassette	£17.95
Mystery Fun House	Cassette	£17.95
Pyramid Fun House	Cassette	£17.95
Pyramid of Doom	Cassette	£17.95
Ghost Town	Cassette	£17.95
Savage Island 1&2	Cassette	£17.95
Golden Voyage	Cassette	£17.95
OTHER APPLICATIONS		
Speech Editor (speech required)	Module	£17.95
Statistics	Module	£44.95
Terminal Emulation II	Module	£44.95
Programming Aids I	Cassette	£11.50
Programming Aids II (contains text to speech)	Disk	£17.95
Programming Aids III	Disk	£17.95

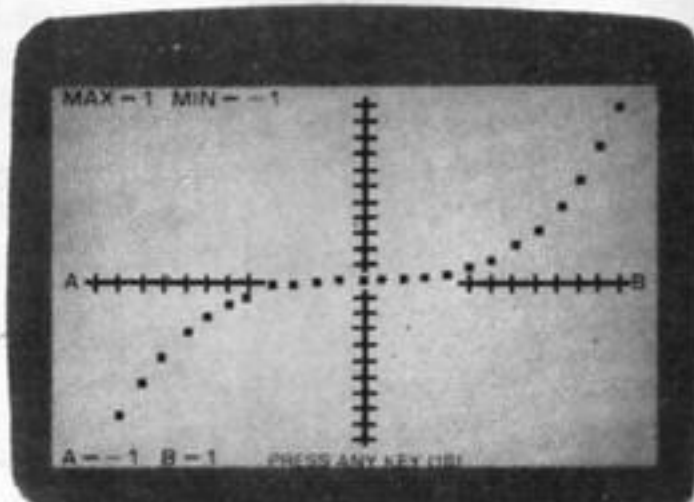
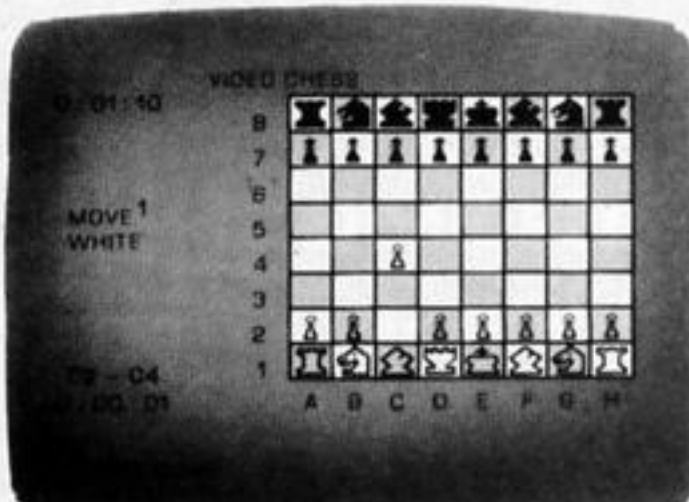
TI HOME COMPUTER HARDWARE

Title	Description	Price inc VAT
TI99/4A	Complete with UHF modulator and power supply	£149.95
PERIPHERALS		
Speech Synthesizer	When used with selected modules will produce electronic speech	£41.95
Peripheral Expansion System	This unit takes all card peripherals and on internal disk drive	£144.95
Disk Drive — Internal	92K formatted drive, mounts internally in Peripheral Expansion System	£199.95
Disk Controller Card	Controls up to 3 Disk Drives, complete with Disk Manager Command Module	£149.95
Disk Drive Double Sided	92K formatted capacity per side acts as 2 drives DSK1 & DSK2 total capacity 184K Bytes	£219.95
Disk Drive External	Complete with own case, power supply & connecting cables	£259.95
RS232 Expansion Card	Provides 2 Serial RS232 Ports, and one parallel port for interfacing	£109.95
RAM Expansion Card	Adds 32K bytes extra RAM bringing total Capacity to 48k bytes	£124.95
P-Code Card	Includes the UCSD-PASCAL P-Code interpreter	£189.95
Matrix Printer	80 Column matrix printer	£219.95
Matrix Printer GP250X	80 column matrix printer with RS232 and centronic parallel interface	£273.95
Epson RX80 Printer	80 column, 100 CPS matrix printer	£320.85
ACCESSORIES		
Joysticks	Wired remote controls	£19.95
Cassette Recorder	Specially designed for computer use	£25.95
Cassette Lead	Allow Connection to cassette recorder	£9.95

We are the official dealers for Texas Instruments, BBC Microcomputer/Acorn Computers, Y.E. Data Business System and Torch Computers. We also stock wide range of printers, monitors and disk drives. Please Phone 0279 443521 for our lowest prices.

TI SOFTWARE

EDUCATIONAL SOFTWARE	Media	Price inc VAT
Beginning Grammar	Module	£17.95
Number Magic	Module	£12.95
Pre-School Fun	Module	£14.50
Hangman	Module	£17.95
Addition/Subtraction 1	Module	£25.95
Addition/Subtraction 2	Module	£25.95
Multiplication	Module	£25.95
Early Reading (Speech required)	Module	£25.95
Music Maker	Module	£25.95
Early Learning Fun	Module	£12.95
Division	Module	£25.95
Aligator Mix	Module	£22.95
Beginners Basic Tutor	Cassette	£8.25
Teach Yourself Extended Basic	Cassette	£11.50
Market Simulation	Cassette	£11.50
Beginners Basic	Manual	£4.95
ORGANISATIONAL SOFTWARE		
Home Budget Management	Module	£22.95
Personel Record Keeping	Module	£34.95
Personel Report Generator	Module	£34.95
Personel Financial Aids	Cassette	£11.50
Inventory	Disk	£69.95
Invoicing	Disk	£69.95
Mailing List	Disk	£69.95
Text Formatter	Module/Disk	£49.90
PROGRAMING LANGUAGES		
Editor/Assembler Language	Module/Disk	£69.95
Extended Basic	Module	£69.95
TI Logo	Module	£69.95
Minimemory	Module	£69.95
TI Writer (Word Processor)	Module	£69.95
UCSD PASCAL		
Editor	Disk	£49.90
Compiler	Disk	£79.95
Linker	Disk	£62.95



AKHTER INSTRUMENTS LTD.

DEP PCN, EXECUTIVE HOUSE, SOUTH ROAD, HARLOW, ESSEX, UK CM20 2BZ
TEL: HARLOW (0279) 412639 OR 443521. TELEX: 995801-A18

ORDERING INFORMATION

All orders which accompany a cheque or cash are **CARRIAGE FREE** (UK ONLY). A carriage charge of 3% of invoice total is applicable to BARCLAYCARD & ACCESS orders. We accept VAT FREE Export orders. Please phone or write for shipping cost. We accept official Government and Education authority orders.

Fire! And it's your chance to be a hero

The firebugs have been at work again. This time it's a tall apartment building burning from the top down.

Your task is to try and catch the unhappy residents as they leap from the windows of the burning building.

My Fireman program runs on the standard TI-99/4A.

Move the firemen and blanket using keys S and D with the alpha-lock button in the off position. Press the shift key first if you need extra speed.

As the fire moves downwards you will have less time in which to prevent the luckless inhabitants from coming to a sticky end.

Catch as many as you can in the blanket, but miss 10 and I'm afraid that's the end of your career with the fire service!

```

100 REM
110 REM FIREMAN
120 REM
130 REM BY VINCE APPS
140 REM
150 GOTO 910
160 REM MOVE SUBROUTINE
170 REM LOOK AT KEYBOARD
180 CALL KEY(0,K,S)
190 OFC=FC
200 IF (K<>115)+(FC<6) THEN 220
210 FC=FC-1
220 IF (K<>100)+(FC>30) THEN 240
230 FC=FC+1
240 IF (K<>83)+(FC<6) THEN 260
250 FC=FC-3
260 IF (K<>68)+(FC>28) THEN 280
270 FC=FC+3
280 IF OFC=FC THEN 340
290 REM PRINT FIREMEN
300 CALL HCHAR(23,2,32,30)
310 CALL HCHAR(23,FC-1,137)
320 CALL HCHAR(23,FC,139)
330 CALL HCHAR(23,FC+1,138)
340 RETURN
350 REM FALLING MAN SUBROUTINE
360 IF JP=1 THEN 450
370 REM NEW MAN
380 CL=INT(RND*18+9)
390 RW=TRW
400 ST=131
410 JP=1
420 NJC=NJC+1
430 IF NJC/5<>INT(NJC/5) THEN 450
440 GOSUB 800
450 RW=RW+1
460 IF RW>24 THEN 570
470 REM MISSED
480 LST=LST+1
490 CALL SOUND(500,220,4,-5,2)

```

Now's your opportunity to prove your skill in the fire service, thanks to Vince Apps and his program for the standard TI-99/4A

```

500 CALL HCHAR(24,CL,146)
510 JP=0
520 M$=STR$(LST)
530 PSN=25
540 GOSUB 740
550 IF LST=10 THEN 1380
560 GOTO 720
570 CALL HCHAR(RW-1,CL,ST)
580 CALL GCHAR(RW,CL,ST)
590 CALL SOUND(100,5280/RW,4)
600 CALL HCHAR(RW,CL,145)
610 IF RW>23 THEN 720
620 CALL HCHAR(RW,CL,ST)
630 IF FC<>CL THEN 720
640 REM SAVED
650 SVD=SVD+1
660 CALL SOUND(300,550,3)
670 CALL HCHAR(23,FC,139)
680 JP=0
690 M$=STR$(SVD)

```

```

700 PSN=12
710 GOSUB 740
720 RETURN
730 REM PRINT@ SUBROUTINE
740 FOR J=1 TO LEN(M$)
750 CH=ASC(SEG$(M$,J,1))
760 CALL HCHAR(1,PSN+J,CH)
770 NEXT J
780 RETURN
790 REM NEW LAYER
800 TRW=TRW+2
810 RW=RW+2
820 IF TRW>22 THEN 1300
830 FOR J=7 TO 28
840 CALL SOUND(100,-5,2)
850 CALL HCHAR(TRW-3,J,32)
860 CALL HCHAR(TRW-2,J,32)
870 CALL HCHAR(TRW-1,J,136)
880 NEXT J
890 RETURN
900 REM PROGRAM START
910 RANDOMIZE
920 CALL CLEAR
930 CALL SCREEN(8)

```

How it works

160-340 move firemen subroutine
180 look at keyboard
190 store old position
200-270 select direction and distance
300-330 print firemen
360-720 falling man subroutine
370-440 new man
450 increment level
480-550 missed man
570-600 print man
650-710 saved man
740-770 simulation of Print at command
800-880 remove layer from building
910 program start
950-1050 define characters and colours
1060-1080 initial values
1100-1170 draw building
1190-1210 print headings
1220-1240 print firemen at start
1260-1280 main program loop
1300-1350 end of game
1380-1420 10 lives lost

Hints on conversion

Texas Basic has some unusual commands for graphics and sound:

CALL CHAR (number, hexadecimal string) controls the user definition of characters. Number refers to the ASCII code of the re-defined character.

CALL HCHAR (row, column, number) prints the character whose ASCII code is number at position row, column on the screen. The TI99/4A has a screen size of 24 rows by 32 columns.

CALL GCHAR (row, column, variable) is the equivalent of PEEK. The ASCII code of the character at position row, column on the screen is left in the designated variable.

CALL SOUND (d, f, v) sounds a note of frequency, f, duration d milli-seconds and volume v.

CALL KEY (O, K, S), the equivalent of INKEY\$, will leave the ASCII code of the key pressed in the variable K.

CALL CLEAR clears the screen. A direct equivalent of CLS.

TI-99/4A PROGRAM

```

940 REM  DEFINE CHARACTERS
950 CALL CHAR(130,"FFFFFFFFFFFFFFFF")
960 CALL CHAR(131,"FF818181818181FF")
970 CALL CHAR(136,"000091AADFEBFFFF")
980 CALL CHAR(137,"3078307CB33148CC")
990 CALL CHAR(138,"0C1E0C3ECD8C1233")
1000 CALL CHAR(139,"000000000081FF")
1010 CALL CHAR(145,"3838917E38382844")
1020 CALL CHAR(146,"0000000138FE7830")
1030 CALL COLOR(13,13,1)
1040 CALL COLOR(14,7,1)
1050 CALL COLOR(15,5,1)
1060 FC=6
1070 OFC=6
1080 TRW=7
1090 REM  DRAW BUILDING
1100 CALL VCHAR(6,7,130,17)
1110 CALL VCHAR(6,28,130,17)
1120 FOR J=6 TO 22 STEP 2
1130 CALL HCHAR(J,8,130,20)
1140 NEXT J
1150 FOR J=7 TO 21 STEP 2
1160 CALL HCHAR(J,8,131,20)
1170 NEXT J
1180 REM  HEADINGS
1190 M$="SAVED:
1200 PSN=5

```

```

1210 GOSUB 740
1220 CALL HCHAR(23,FC-1,137)
1230 CALL HCHAR(23,FC,139)
1240 CALL HCHAR(23,FC+1,138)
1250 REM  MAIN LOOP
1260 GOSUB 180
1270 GOSUB 360
1280 GOTO 1260
1290 REM  SUCCESS

```

```

1300 CALL SOUND(500,330,2)
1310 CALL SOUND(500,450,2)
1320 CALL SOUND(500,360,2)
1330 CALL SOUND(900,500,2)
1340 PRINT "CONGRATULATIONS!"
1350 PRINT "LIVES SAVED:";SVD
1360 GOTO 1430
1370 REM  FAILURE
1380 FOR J=550 TO 110 STEP -20
1390 CALL SOUND(150,J,2)
1400 NEXT J
1410 PRINT "10 LIVES LOST - DEMOTION TO"
1420 PRINT "BRASS POLISHER, 3RD CLASS!"
1430 END

```

LOST:"

AT LAST!!
SINCLAIR
SPECTRUM
JOYSTICK
CONTROLLER

£14.95!!

ALL PRICES INCLUDE V.A.T. & P. & P.

LARGE RANGE OF
 COMPONENTS, HARDWARE
 AND SOFTWARE

THE I²L PLEDGE - IF WE CAN'T DESPATCH WITHIN 5 WORKING DAYS
 OF RECEIVING YOUR ORDER - WE WILL RETURN YOUR MONEY - AND
 THAT'S A PROMISE.

SPECTRUM
SOUND
AMPLIFIER
WITH SPEAKER
MAINS POWERED
WITH SAVE/LOAD
SWITCHING
ONLY £8.95

SPECTRUM
MEMORY
UPGRADES
AN EXTRA 32K FOR
ONLY
£24.95 FOR TYPE 2
MACHINES

I²L
 INTERACTIVE
 INSTRUMENTS LTD
 INTERACTIVE HOUSE
 GREAT CENTRAL ST
 LEICESTER
 LEIC. 532167

DISK DRIVES
THE LATEST
SHUGART SA200
40 TRACK DRIVES
COMPLETE WITH
POWER SUPPLY
AND STEEL CASE
FOR ONLY
£199
FOR BBC

You're the only one who can save our planet

Well, it had to happen. All those alien invaders finally took over the Earth, in spite of a spirited defence on computers throughout the land.

Having grown fat (and red) on a diet of human blood they've now retired to their strongly defended base somewhere in England's green and pleasant countryside, leaving you with only a single tank to your name.

My program, which runs in 16K on the standard TI-99/4A, is called Last Stand and simulates the final battle.

Move with the arrow keys (ES DX), with the alpha-lock button down. Fire with the space-bar.

Your task, in the unlikely event of your survival, is to destroy the alien headquarters at the top of the screen.

Luckily the aliens are pretty dumb and you will be able to hide behind trees and shoot from under cover, until, that is, you run out of trees. . .

It's up to you now, so climb into your tank and tackle the alien invaders in Vince Apps' program for the standard TI-99/4A

How it works

160-800 move tank subroutine
170 look at keyboard
200-580 fire shell
600-610 store row and column positions
620-730 select direction and new position
750-790 draw tank
820-1020 move alien subroutine
820-860 generate new alien
870-880 store row and column positions
890-960 select new position
970-1010 print alien
1040 start of main program
1130-1270 define characters and colours
1290-1370 initial values
1390-1480 print screen at start
1500-1530 main subroutine calling loop
1550-1560 delay subroutine
1590-1630 destroyed by alien
1660-1760 HQ hit

```
100 REM
110 REM LAST STAND
120 REM
130 REM BY VINCE APPS
140 REM
150 GOTO 1040
160 REM MOVE TANK SUBROUTINE
170 CALL KEY(O,K,S)
180 IF K<>32 THEN 600
190 REM FIRE SHELL
200 ON D GOTO 210,230,250,270
210 R=-1
220 GOTO 440
```

```
230 R=1
240 GOTO 440
250 C=-1
260 GOTO 280
270 C=1
280 FOR J=CL TO CL+C*5 STEP C
290 IF (J>31)+(J<2) THEN 420
300 CALL GCHAR(RW,J,X)
310 CALL HCHAR(RW,J,140)
320 CALL SOUND(100,900,4)
330 CALL HCHAR(RW,J,X)
340 IF (X=128)+(X=129) THEN 1660
350 IF X<>130 THEN 420
360 CALL SOUND(350,-5,2)
370 CALL SCREEN(7)
380 GOSUB 1550
390 CALL SCREEN(4)
400 CALL HCHAR(RW,J,32)
410 GOTO 800
420 NEXT J
430 GOTO 800
440 FOR J=RW TO RW+R*5 STEP R
450 IF (J>23)+(J<1) THEN 580
460 CALL GCHAR(J,CL,X)
470 CALL HCHAR(J,CL,140)
480 CALL SOUND(100,900,4)
490 CALL HCHAR(J,CL,X)
500 IF (X=128)+(X=129) THEN 1660
510 IF X<>130 THEN 580
520 CALL SOUND(350,-5,2)
530 CALL SCREEN(7)
540 GOSUB 1550
550 CALL SCREEN(4)
560 CALL HCHAR(J,CL,32)
570 GOTO 800
```

```
580 NEXT J
590 GOTO 800
600 OCL=CL
610 ORW=RW
620 IF (K<>69)+(RW<3) THEN 650
630 D=1
640 RW=RW-1
650 IF (K<>88)+(RW>22) THEN 680
660 D=2
670 RW=RW+1
680 IF (K<>83)+(CL<3) THEN 710
690 D=3
```

Hints on conversion

Texas Basic has some unusual commands for graphics and sound:

CALL CHAR (number, hexadecimal string) controls the user definition of characters. Number refers to the ASCII code of the re-defined character.

CALL HCHAR (row, column, number) prints the character whose ASCII code is number at position row, column on the screen. The TI99/4A has a screen size of 24 rows by 32 columns.

CALL GCHAR (row, column, variable) is the equivalent of PEEK. The ASCII code of the character at position row, column on the screen is left in the designated variable.

CALL SOUND (d, f, v) sounds a note of frequency f, duration d milli-seconds and volume v.

CALL KEY (O, K, S), the equivalent of INKEY\$, will leave the ASCII code of the key pressed in the variable K.

CALL CLEAR clears the screen. A direct equivalent of CLS.

TI-99/4A PROGRAM

```

700 CL=CL-1
710 IF (K<>68)+(CL>30) THEN 740
720 D=4
730 CL=CL+1
740 IF (OCL=CL)*(ORW=RW) THEN 800
750 CALL HCHAR(ORW,OCL,ST)
760 CALL GCHAR(RW,CL,ST)
770 CALL SOUND(50,440,3)
780 IF (ST=144)+(ST=152) THEN 800
790 CALL HCHAR(RW,CL,120+D)
800 RETURN
810 REM MOVE ALIEN
820 CALL GCHAR(ER,EC,X)
830 IF X=130 THEN 870
840 ER=2
850 EC=INT(RND*30)+1
860 ES=32
870 OER=ER
880 OEC=EC
890 IF ER>=RW THEN 920
900 ER=ER+1
910 GOTO 970
920 ER=ER-1
930 IF EC>=CL THEN 960
940 EC=EC+1
950 GOTO 970
960 EC=EC-1
970 CALL HCHAR(OER,OEC,ES)
980 CALL GCHAR(ER,EC,ES)
990 IF (ES<125)*(ES>120) THEN 1590
1000 CALL SOUND(50,220,4,-3,4)
1010 CALL HCHAR(ER,EC,130)
1020 RETURN
1030 REM PROGRAM START
1040 CALL CLEAR
1050 RANDOMIZE
1060 PRINT "SKILL LEVEL":
1070 PRINT "1=EASY":
1080 PRINT "10=DIFFICULT":
1090 INPUT SK
1100 IF (SK>10)+(SK<1) THEN 1090
1110 CALL CLEAR
1120 REM DEFINE CHARACTERS
1130 CALL CHAR(121,"1054547C7C7C5444")
1140 CALL CHAR(122,"222A3E3E3E2A2A08")
1150 CALL CHAR(123,"00007F1CFE1CF")
1160 CALL CHAR(124,"00FE387F38FE")
1170 CALL CHAR(128,"8080C1C3C7CFDFFF")
1180 CALL CHAR(129,"010183C3E3F3FBFF")
1190 CALL CHAR(130,"C3FFDBFFFC3BD1866")
1200 CALL CHAR(131,"00000034FDFFFFFF")
1210 CALL CHAR(140,"0000001818")
1220 CALL CHAR(144,"183C7EFFFF7F7E3C")
1230 CALL CHAR(152,"1818181818183C")
1240 CALL COLOR(12,5,1)
1250 CALL COLOR(13,7,1)
1260 CALL COLOR(15,13,1)
1270 CALL COLOR(16,11,1)
1280 REM INITIAL VALUES
1290 RW=22

```

```

1300 ORW=22
1310 CL=17
1320 OCL=17
1330 ST=32
1340 D=1
1350 ER=2
1360 EC=INT(RND*20)+5
1370 ES=32
1380 REM PLANT TREES
1390 FOR K=1 TO 60-SK*5
1400 RR=INT(RND*22)+2
1410 RC=INT(RND*29)+3
1420 CALL HCHAR(RR,RC,152)
1430 CALL HCHAR(RR-1,RC,144)
1440 NEXT K
1450 CALL HCHAR(RW,CL,121)
1460 CALL HCHAR(ER,EC,130)
1470 CALL HCHAR(1,16,128)
1480 CALL HCHAR(1,17,129)
1490 REM MAIN LOOP
1500 GOSUB 820
1510 GOSUB 170
1520 GOSUB 820
1530 GOSUB 820
1540 GOTO 1500
1550 FOR DE=1 TO 100
1560 NEXT DE
1570 RETURN
1580 REM FAILURE
1590 FOR K=1 TO 10
1600 CALL HCHAR(RW,CL,32)
1610 CALL SOUND(200,-5,1)
1620 CALL HCHAR(RW,CL,120+D)
1630 NEXT K
1640 GOTO 1770
1650 REM SUCCESS
1660 CALL SOUND(300,-5,2)
1670 CALL HCHAR(1,16,131)
1680 CALL HCHAR(1,17,131)
1690 CALL SOUND(500,-5,2)
1700 FOR DE=1 TO 1000
1710 NEXT DE
1720 CALL SOUND(500,330,2)
1730 CALL SOUND(500,450,2)
1740 CALL SOUND(500,360,2)
1750 CALL SOUND(900,500,2)
1760 PRINT "SUCCESS!":
1770 INPUT "PLAY AGAIN? ":Q$
1780 IF SEG$(Q$,1,1)="Y" THEN 1040
1790 END

```



Program power for you and your TI-99/4A

Starship Pegasus £11.00

By Not Polyoptics from Stainless Software, 10 Alstone Road, Stockport, Cheshire, SK4 5AH

I doubt if you will come across a more diverse and outrageous collection of life forms as those you will encounter in command of the Starship Pegasus. Your mission is to make treaties with these creatures in order to exploit their planets for as many energy, technology and mineral points as you can obtain.

If they prove unfriendly and are lesser beings you may take a chance and open fire on their planet, exploiting it after their destruction. Underestimate their intelligence, however, and you may find yourself under attack!

For the seasoned campaigner there are over 100 quadrants to explore, one quadrant being chosen for each game. All have their own star systems which in turn have their own planets. Some of these will be uninhabited and make easy pickings.

Your journey is not

By popular demand, we take a look at a selection of games available now for the TI-99/4A. Read on for our reviewers' verdicts — and star ratings

necessarily a smooth one as space marauders are always waiting to launch an attack. These make frustratingly evasive opponents in an exchange of fire.

Graphics are not a main feature of this program, but where used are effective.

If you survive your mission you may be awarded a medal — but not always, as my husband found out when he beamed down to a planet, was granted immortality but then tortured for eons!

J.W.

instructions	80%
playability	70%
graphics	30%
value for money	60%



Bluegrass Sweepstakes £7

By Pewterware from Stainless Software, 10 Alstone Road, Stockport, Cheshire SK4 5AH

Whether you enjoy the occasional flutter on the gee-gees or not, you will like this horseracing game for up to eight players.

Eight horses are displayed on screen together with their names and starting prices. Bets are placed and the race begins. The eight horses move across the screen from left to right and winnings are paid out on the

first, second and third horses across the line.

Each punter enters his or her name and the computer will then display the name and the amount that person has available to bet with prior to each race. You are given 1000 dollars to begin with. The winner of each race receives a sweepstake ticket in addition to any winnings and this is used in the ninth race in which the runners are the winners of the previous eight races.

The graphics and colour capabilities of the TI99/4A are used to excellent effect in this very well-thought-out program.

It is not a game where you continually involved in the action, but it is nevertheless a very entertaining programme for children and adults alike.

D.B.

instructions	70%
playability	70%
graphics	100%
value for money	80%



Astro Fighter £3.50

Christine Computing, 6 Florence Close, Watford, Herts.

The aim of the game is to shoot down X-wing fighters by aligning them in your sights, using either keyboard or joysticks. Sufficient X-wings must be shot down to prevent your own craft running out of fuel.

It's confusing to play at first, since the controls actually move the enemy fighter, not your craft. The stars stay fixed.

In-program instructions were adequate, and the game seemed robust and well tested.

Inevitably, though, a game written in TI Basic is going to be very slow and rather boring to play.

D.G.

instructions	80%
playability	50%
graphics	50%
value for money	40%



Pen the Pig £4

Stainless Software, 10 Alstone Road, Stockport, Cheshire SK4 5AH

The idea of this game is to trap a pig running around in a large pen and to contain it into one of two smaller pens located in two opposite corners.

The pig is represented by a bouncing ball, and you can "draw" walls in order to contain it by using the four direction keys, E,X,D and S. Occasionally the pig will break through the wall you have drawn.

The program loads easily and there are two skill levels. At level one the game proves to be quite easy. But at level two you draw an invisible shield on the screen instead of a wall, and this time the game becomes very much harder, as you try to

remember the location of your shield.

Such instructions as are necessary are printed on screen at the beginning of the game and are adequate for this rather uncomplicated programme.

All in all I found the game

rather boring, with elementary use of graphics and sound.

D.B.

instructions	50%
playability	50%
graphics	30%
value for money	20%



Can you tunnel through the rock in time?

You're on an errand of mercy in Dave Slinn's game for the TI-99/4A. So don't delay — it's a matter of life and death.

Deep underground, under four layers of rock, there's a trapped miner — and your job is to rescue him. But there's no time to lose, for any minute now the mine could blow up.

Once you think you've become an expert rescuer, you can increase the challenge by altering line 1720.

How it works

10-220 instructions
300-530 define characters
560-1000 set up playing area
1010-1480 set up random layers of 4 different layers of rock
1490-1610 explosion sound & graphics
1720 initial start time
1730-1800 start of countdown
1840-1870 test for direction of movement
1880-2010 update position of rescuer
2020-2120 keep rescuer in playing area
2140-2350 check future position of rescuer
2370-2700 update position of rescuer, colour flash and drill sound
2710-2810 check old position of rescuer
2820-3100 determine length of time to drill rock
3220-3290 countdown routine
3310-3360 message print routine.

Variables

R new row position of rescuer
C new column position of rescuer
RR old row position of rescuer
CC old column position of rescuer

```

5 REM MINE MAZE D.H.SLINN.
10 CALL CLEAR
20 PRINT "      ****MINE MAZE****"
30 PRINT
40 PRINT "YOU HAVE TO RACE AGAINST"
50 PRINT "TIME THRO' DIFFERENT LAYERS"
60 PRINT "OF ROCK TO SAVE THE MINER."
70 PRINT
80 PRINT "  THERE ARE 4 DIFFERENT"
90 PRINT "HARDNESS'S OF ROCK RANDOMLY"
100 PRINT "PLACED. AS EACH SECTION IS"
110 PRINT "DRILLED IT'S HARDNESS AND"
120 PRINT "THE POSITION OF ROCK OF THE"
130 PRINT "SAME HARDNESS IS SHOWN FOR A"
140 PRINT "      SHORT TIME"
150 PRINT
160 PRINT "SOME OF THE ROCK IS TO HARD"
170 PRINT "      FOR YOUR DRILL."
180 PRINT
190 PRINT "      ****USE ARROW KEYS****"
200 PRINT
210 PRINT
220 PRINT "      PRESS ANY KEY TO START."
230 PRINT
240 CALL KEY(0,K,S)
250 IF S=0 THEN 240
260 CALL CLEAR
270 FOR I=9 TO 16
280 CALL COLOR(I,2,2)
290 NEXT I
295 REM DEFINE GRAPHICS
300 CALL CHAR(128,"3C18181818181818")
310 CALL CHAR(129,"1C18101F1C1C1476")
320 CALL CHAR(130,"03071F3F7F7FFFFF")
330 CALL CHAR(131,"FFFFFFFFFFFFFFFF")
340 CALL CHAR(132,"1818181818181818")
350 CALL CHAR(133,"FFA5A5A5A5A5A5FF")
360 CALL CHAR(134,"80C0C0E0F9F8FCFF")
370 CALL CHAR(135,"0")
380 CALL CHAR(136,"3F40FF4040585858")
390 CALL CHAR(137,"FC02FF02FAAFAA02")
400 CALL CHAR(138,"3F40DF55555F4040")
410 CALL CHAR(139,"FC02F8A8A8FD0202")
420 CALL CHAR(140,"FF010A5B08181818")
430 CALL CHAR(141,"3C42A59999A5423C")
440 CALL CHAR(142,"18242424243C5A5A")
450 CALL CHAR(143,"5A99999999999999")
460 CALL CHAR(144,"FFFFFFFFFFFFFFFF")
470 CALL CHAR(145,"030F1F3F3F7FFFFF")
480 CALL CHAR(146,"0000030F1F1F7FFF")
490 CALL CHAR(147,"000000C0C0E0F8FF")
500 CALL CHAR(148,"80C0C0E0F8F8FCFF")
510 CALL CHAR(149,"00000000E0FFFF22")
520 CALL CHAR(150,"18183C5A5A18183C")
530 CALL CHAR(152,"FFC3A49999A542FF")
540 CALL CLEAR
550 CALL SCREEN(6)
560 CALL HCHAR(4,11,141)
570 CALL HCHAR(5,11,142)
580 CALL HCHAR(6,12,136)
590 CALL HCHAR(6,13,137)
600 CALL HCHAR(6,15,138)
610 CALL HCHAR(6,16,139)
620 CALL HCHAR(6,19,145)
630 CALL HCHAR(6,20,144)
640 CALL HCHAR(6,21,144)

```

Hints on conversion

TI Basic is almost the same as other micro basics with the following exceptions

CALL HCHAR pokes the screen with the character named eg CALL HCHAR (12,6,32) puts a space at row 12 column 6

eg CALL HCHAR (12,6,32, 8) will put 8 spaces horizontally across the screen starting at row 12 column 6.

CALL VCHAR same as HCHAR but vertical

CALL CHAR defines user graphics

CALL GCHAR peeks the screen, same format as HCHAR

CALL KEY inputs data direct from the keyboard without having to press return/enter key

CALL CLEAR clears the screen
The TI screen is divided into 32 vertical columns and 24 horizontal rows.

```

650 CALL HCHAR(6,22,148)
660 CALL HCHAR(5,20,146)
670 CALL HCHAR(5,21,147)
680 CALL HCHAR(6,14,140)
690 CALL HCHAR(6,11,143)
700 CALL HCHAR(17,1,131)
710 CALL HCHAR(17,2,134)
720 CALL VCHAR(7,1,120,18)
730 FOR I=3 TO 10
740 CALL HCHAR(17,I,135)
750 NEXT I
760 CALL VCHAR(23,11,131,2)
770 CALL VCHAR(7,11,132,15)
780 CALL HCHAR(22,11,133)
790 CALL HCHAR(17,3,135,8)
800 FOR I=3 TO 9 STEP 2
810 CALL HCHAR(17,I,128)
820 NEXT I
830 CALL HCHAR(6,1,32,10)
840 CALL VCHAR(7,11,132,15)
850 CALL HCHAR(6,17,32,2)
860 CALL HCHAR(6,23,32,10)
870 CALL HCHAR(6,19,145)
880 CALL HCHAR(6,20,131,2)
890 CALL HCHAR(6,22,148)
900 FOR I=13 TO 27 STEP 2
910 CALL HCHAR(22,I,128)
920 NEXT I
930 FOR I=12 TO 26 STEP 2
940 CALL HCHAR(22,I,135)
950 NEXT I
960 CALL HCHAR(22,29,130)
970 CALL HCHAR(22,30,131,3)
980 FOR I=3 TO 9 STEP 2
990 CALL HCHAR(17,I,128)
1000 NEXT I

```


TI-99/4A PROGRAM



```

1005 REM ' POSITION DIFFERENT LAYERS OF ROCK
1010 RANDOMIZE
1020 C=12
1030 R=7
X 1040 A=INT(4*RND)+12
1050 CALL HCHAR(R,C,A*8,3)
1060 C=C+3
1070 IF C>32 THEN 1080 ELSE 1040
1080 C=12
1090 R=R+1
1100 IF R=22 THEN 1090 ELSE 1110
1110 IF R=25 THEN 1120 ELSE 1040
1120 C=2
1130 R=7
1140 A=INT(4*RND)+12
1150 CALL HCHAR(R,C,A*8,3)
1160 C=C+3
1170 IF C>10 THEN 1180 ELSE 1140
1180 C=2
1190 R=R+1

```

```

1200 IF R=17 THEN 1210 ELSE 1220
1210 R=R+1
1220 IF R=25 THEN 1230 ELSE 1140
1230 FOR I=1 TO 3
1240 C=INT(17*RND)+13
1250 CALL VCHAR(7,C,96,10)
1260 NEXT I
1270 FOR I=1 TO 3
1280 C=INT(9*RND)+2
1290 CALL VCHAR(8,C,96,7)
1300 NEXT I
1310 FOR I=1 TO 4
1320 R=INT(14*RND)+7
1330 C=INT(7*RND)+2
1340 CALL HCHAR(R,C,152,3)
1350 CALL VCHAR(R,C+2,152,2)
1360 NEXT I
1370 FOR I=1 TO 12
1380 R=INT(16*RND)+7
1390 C=INT(19*RND)+12
1400 CALL HCHAR(R,C,152,2)
1410 CALL VCHAR(R,C+1,152,3)
1420 NEXT I
1430 CALL HCHAR(22,28,129)
1440 CALL COLOR(13,2,15)
1450 CALL VCHAR(7,1,120,18)
1460 CALL HCHAR(17,2,134)
1470 CALL COLOR(14,15,6)
1480 CALL COLOR(15,2,6)
1485 REM EXPLOSION
1490 CALL SOUND(1000,-4,0)
1500 CALL SOUND(100,-7,0)
1510 CALL HCHAR(22,26,134)
1520 CALL SOUND(100,-6,0)
1530 CALL HCHAR(22,25,144)
1540 CALL SOUND(50,-7,2)
1550 CALL HCHAR(22,18,144,7)
1560 CALL SOUND(600,-6,2)
1570 CALL HCHAR(22,17,130)
1580 CALL SOUND(1000,-7,0)
1590 CALL HCHAR(22,15,134)
1600 CALL HCHAR(22,14,130)
1610 CALL HCHAR(22,11,134)
1620 FOR I=1 TO 3
1630 CALL HCHAR(6,2,150)
1640 FOR D=1 TO 200
1650 NEXT D
1660 CALL HCHAR(6,2,32)
1670 FOR D=1 TO 100
1680 NEXT D
1690 CALL SOUND(50,1000,5)
1700 NEXT I

```

```

1710 CALL HCHAR(6,2,150)
1715 REM SET UP START OF COUNT
1720 TIME=90
1730 R=6
1740 C=2
1750 TIME=TIME-1
1760 IF LEN(STR$(TIME))=1 THEN 1770 ELSE 1780
1770 CALL HCHAR(4,26,32)
1780 FOR I=1 TO LEN(STR$(TIME))
1790 CALL HCHAR(4,I+24,ASC(SEG$(STR$(TIME),I,1)))
1800 NEXT I
1810 CALL SOUND(100,-4,2)
1820 CALL KEY(0,K,S)
1825 REM TEST FOR 'OUT OF TIME'
1830 IF TIME=0 THEN 3380
1840 IF K=68 THEN 1900 ELSE 1850
1850 IF K=83 THEN 1920 ELSE 1860
1860 IF K=69 THEN 1940 ELSE 1870
1870 IF K<>88 THEN 1750 ELSE 1880
1880 DR=+1
1890 GOTO 1950
1900 DC=+1
1910 GOTO 1990
1920 DC=-1
1930 GOTO 1990
1940 DR=-1
1950 RR=R
1960 CC=C
1970 R=R+DR
1980 GOTO 2020
1990 CC=C
2000 RR=R
2010 C=C+DC
2020 IF C>32 THEN 2030 ELSE 2050
2030 C=32
2040 GOTO 1750
2050 IF C<1 THEN 2060 ELSE 2080
2060 C=1
2070 GOTO 1750
2080 IF R<6 THEN 2090 ELSE 2110
2090 R=6
2100 GOTO 1750
2110 IF R>24 THEN 2120 ELSE 2140
2120 R=24
2130 GOTO 1750
2135 REM TEST FOR HARDNESS OF ROCK
2140 CALL GCHAR(R,C,X)
2150 IF X=96 THEN 2370
2160 IF X=104 THEN 2450
2170 IF X=112 THEN 2560
2180 IF X=120 THEN 2820
2190 IF X=32 THEN 2920
2200 IF X=131 THEN 2950
2210 IF X=128 THEN 2990
2220 IF X=134 THEN 3020
2230 IF X=144 THEN 3020
2240 IF X=130 THEN 3020
2250 IF X=148 THEN 3020

```


TI-99/4A PROGRAM

```

2260 IF X=145 THEN 3020
2270 IF X=135 THEN 2990
2280 IF X=129 THEN 3060
2290 IF X=143 THEN 3460
2300 IF X=132 THEN 3480
2310 IF X=136 THEN 3550
2320 IF X=137 THEN 3550
2330 IF X=138 THEN 3550
2340 IF X=139 THEN 3550
2350 IF X=152 THEN 2660
2360 GOTO 1750
2370 CALL COLOR(9,11,11)
2380 CALL SOUND(200,30000,30,30000,30,440,30,-4,0)
2390 CALL SOUND(400,30000,30)

```

```

2960 CALL HCHAR(RR,CC,135)
2970 CALL HCHAR(R,C,150)
2980 GOTO 1750
2990 CALL HCHAR(RR,CC,135)
3000 CALL HCHAR(R,C,150)
3010 GOTO 1750
3020 CALL SOUND(100,110,9,140,9,440,30,-4,0)
3030 CALL HCHAR(RR,CC,135)
3040 CALL HCHAR(R,C,150)

```

```

2400 CALL COLOR(9,2,2)
2410 GOSUB 3220
2420 CALL HCHAR(RR,CC,135)
2430 CALL HCHAR(R,C,150)
2440 GOTO 1750
2450 REM
2460 FOR D=1 TO 2
2470 CALL COLOR(10,10,10)
2480 CALL SOUND(200,30000,30,440,12,440,30,-4,0)
2490 CALL SOUND(400,30000,30)
2500 CALL COLOR(10,2,2)
2510 GOSUB 3220
2520 NEXT D
2530 CALL HCHAR(RR,CC,135)
2540 CALL HCHAR(R,C,150)
2550 GOTO 1750
2560 FOR D=1 TO 3
2570 CALL COLOR(11,5,5)
2580 CALL SOUND(200,30000,30,880,12,440,30,-4,0)
2590 CALL SOUND(400,30000,30)
2600 CALL COLOR(11,2,2)
2610 GOSUB 3220
2620 NEXT D
2630 CALL HCHAR(RR,CC,135)
2640 CALL HCHAR(R,C,150)
2650 GOTO 1750
2660 FOR D=1 TO 3
2670 CALL COLOR(16,16,2)
2680 CALL SOUND(200,30000,30,1000,12,440,30,-4,0)
2690 CALL COLOR(16,2,2)
2700 NEXT D
2710 IF K=68 THEN 2760
2720 IF K=83 THEN 2780
2730 IF K=69 THEN 2800
2740 R=R-1
2750 GOTO 1750
2760 C=C-1
2770 GOTO 1750
2780 C=C+1
2790 GOTO 1750
2800 R=R+1
2810 GOTO 1750
2820 FOR D=1 TO 4
2830 CALL COLOR(12,4,4)
2840 CALL SOUND(150,30000,30,880,12,440,30,-4,0)
2850 CALL SOUND(400,30000,30)
2860 CALL COLOR(12,2,2)
2870 GOSUB 3220
2880 NEXT D
2890 CALL HCHAR(RR,CC,135)
2900 CALL HCHAR(R,C,150)
2910 GOTO 1750
2920 CALL HCHAR(RR,CC,32)
2930 CALL HCHAR(R,C,150)
2940 GOTO 1750
2950 CALL SOUND(100,200,4,210,4,-4,3)

```

```

3050 GOTO 1750
3060 FOR I=1 TO 4
3070 CALL SOUND(100,1000,2)
3080 FOR D=1 TO 50
3090 NEXT D
3100 NEXT I
3105 REM WINNING ROUTINE
3110 MS="YOU DID IT!"
3120 R=18
3130 C=19
3140 GOSUB 3310
3150 MS="PRESS ANY KEY"
3160 R=20
3170 C=18
3180 GOSUB 3310
3190 CALL KEY(0,K,S)
3200 IF S=0 THEN 3190
3210 GOTO 780
3215 REM TIME COUNTDOWN ROUTINE
3220 TIME=TIME-1
3230 IF LEN(STR$(TIME))=1 THEN 3240 ELSE 3250
3240 CALL HCHAR(4,26,32)
3250 FOR I=1 TO LEN(STR$(TIME))
3260 CALL HCHAR(4,I+24,ASC(SEG$(STR$(TIME),I,1)))
3270 NEXT I
3280 CALL SOUND(100,-4,2)
3290 IF TIME=0 THEN 3380
3300 RETURN
3305 REM PRINT MESSAGE ROUTINE
3310 FOR I=1 TO LEN(MS)
3320 CD=ASC(SEG$(MS,I,1))
3330 CALL HCHAR(R,C,CD)
3340 C=C+1
3350 NEXT I
3360 RETURN
3370 STOP
3375 REM LOSE ROUTINE
3380 CALL SOUND(100,200,4)
3390 CALL SOUND(250,110,1)
3400 MS="***FAILED***"
3410 R=16
3420 C=17
3430 GOSUB 3310
3440 GOTO 3150
3450 END
3460 C=10
3470 GOTO 1750
3480 FOR D=1 TO 4
3490 CALL SOUND(200,400,4,200,4,-4,2)
3500 GOSUB 3220
3510 NEXT D
3520 CALL HCHAR(RR,CC,135)
3530 CALL HCHAR(R,C,150)
3540 GOTO 1750
3550 R=7
3560 GOTO 1750

```


How's your memory? It needs to be good to escape from the maze

How it works

40-100 define characters
120-170 set colours
200-370 lays out screen, prints score, GOSUB to walls lay out
380-400 print indicator showing level
430-460 wait for key press, check if maze finished
480-590 check E pressed, test for fruit ahead, move upwards and beep
610-720 as above, except key X, move down
740-840 key S pressed. Check for invisible door, moves through if there is GOSUBS score increment, or decrement if no door.
860-910 routine for printing on screen
930-1170 set flash time according to level draw walls and places random doors
1180-1340 randomise to see of to print fruit check if fruit all used if not, GOSUB print fruit.
1270-1340 Print fruit.
1360-1450 Effects for eating fruit increase score
1470-1580 flash openings in wall for set time
1600-1700 effects for moving through opening
1720-1830 same for failing to find opening correctly
1850-1980 game finished, check score
2000-2150 game over, replay opportunity
2170-2260 new maze, score good enough
2280-2340 variables set-up

In this game, you have to guide a figure through a maze — but it's not as easy as it sounds!

A number of vertical walls are printed down the screen, with channels in between. Random doors are set into each wall.

These openings in the walls flash for a set period of time at the beginning of the game, during which you must memorise their locations.

After 15 seconds (on the easiest level) the doors close up. You must now guide your figure across the screen using keys S and X to move up and down and E to try to move through the now invisible holes in the walls.

The figure leaves a trail behind him. For each door you

You've got just 15 seconds to memorise the location of the doors in John Powers's game for the standard TI-99/4A. And time gets shorter as your memory improves

Hints on conversion

If you would like to try to convert this program to another machine, remember that the Texas has a 24 by 32 screen, and a good amount of space can be saved by using multiple line statements — the program only takes up 5½K anyway — not available on the standard TI.

Here are the more unusual commands:

CALL KEY(M,K,S) Where K is the ASCII value of the key pressed, and S returns 1 if pressed. You can replace this with INKEY or GET.

CALL SOUND(D,N,V) Where D is duration, N is note in Hertz, V is the volume. Most machines have some sound capabilities — so it's easier to write your own.

CALL CHAR(A,NS) Defines characters in Hexadecimal. N is the ASCII character to be redefined. Characters are defined left-to-right from the top. Each digit represents the pattern in a 1×4 horizontal block of pixels; so there are 2 Hex digits for each line of the 8×8 square.

CALL HCHAR(R,C,A,P) Positions ASCII character A, and repeats horizontally from coordinates C and R P times.

CALL VCHAR(R,C,A,P) As above except repeats vertically.

CALL COLOUR(S,F,B) Defines character foreground and background colour. Replace with GCOL or INK or other command.

CALL SCREEN(X) Sets screen colour. Replace with PAPER or VDU 19 or COLOUR statement.

CALL CLEAR Clears screen. Use CLS.

CALL GCHAR(R,C,V) Returns ASCII code of character at position R,C in variable V. Use a POINT or screen PEEK.

successfully pass through, your score increases by 100. If you try to pass through where there is no door you lose 35 points.

When you eat a fruit (placed randomly in the maze), the doors become visible for five seconds and you score bonus points.

There are three levels of skill. An indicator at top left

Variables

Z row of print position
V columns of print position
G level
CL column of figure
RW row of figure
GC ASCII code from GCHAR
CT column position of walls
RA/RB position of doors in walls
COUNT number of fruit on screen
BF number of fruit available
FI random position of fruit
SC score
MS screen messages

shows you current level, you pass onto the next level automatically if your score is high enough. The game is fairly easy to play at the lowest level but it takes a good deal of skill to navigate your way around the harder mazes.

```
5 REM MINE MAZE D.H.SLINN.
10 CALL CLEAR
20 PRINT " *****MINE MAZE*****"
30 PRINT
40 PRINT "YOU HAVE TO RACE AGAINST"
50 PRINT "TIME THRO DIFFERENT LAYERS"
60 PRINT "OF ROCK TO SAVE THE MINER."
70 PRINT
80 PRINT " THERE ARE 4 DIFFERENT"
90 PRINT "HARDNESS S OF ROCK RANDOMLY"
100 PRINT "PLACED. AS EACH SECTION IS"
110 PRINT "DRILLED IT S HARDNESS AND"
120 PRINT "THE POSITION OF ROCK OF THE"
130 PRINT "SAME HARDNESS IS SHOWN FOR A"
140 PRINT " SHORT TIME"
150 PRINT
160 PRINT "SOME OF THE ROCK IS TOO HARD"
170 PRINT " FOR YOUR DRILL "
180 PRINT
190 PRINT " *****USE ARROW KEYS*****"
200 PRINT
210 PRINT
220 PRINT " PRESS ANY KEY TO START "
230 PRINT
240 CALL KEY(O,K,S)
250 IF S=0 THEN 240
260 CALL CLEAR
270 FOR I=9 TO 16
280 CALL COLOR(I,2,2)
290 NEXT I
295 REM DEFINE GRAPHICS
300 CALL CHAR(128,"3C1B1B1B1B1B1B1B")
310 CALL CHAR(129,"1C1B101F1C1C1476")
320 CALL CHAR(130,"03071F3F7F7FFFFF")
330 CALL CHAR(131,"FFFFFFFFFFFFFFFF")
340 CALL CHAR(132,"1B1B1B1B1B1B1B1B")
350 CALL CHAR(133,"FFA5A5A5A5A5A5FF")
360 CALL CHAR(134,"80C0C0E0F9F8FCFF")
370 CALL CHAR(135,"0")
380 CALL CHAR(136,"3F40FF40405B5B5B")
390 CALL CHAR(137,"FC02FF02FAAFA02")
400 CALL CHAR(138,"3F40DF55555F4040")
410 CALL CHAR(139,"FC02FBAAAFD0202")
```


TI-99/4A PROGRAM

```

420 CALL CHAR(140,"FF81BDA5BD818181")
430 CALL CHAR(141,"3C42A59999A5423C")
440 CALL CHAR(142,"18242424243C5A5A")
450 CALL CHAR(143,"5A99999999999999")
460 CALL CHAR(144,"FFFFFFFFFFFFFFFF")
470 CALL CHAR(145,"030F1F3F3F7FFFFF")
480 CALL CHAR(146,"0000030F1F1F7FFF")
490 CALL CHAR(147,"000080C0C0E0F8FF")
500 CALL CHAR(148,"80C0C0E0F8F8FCFF")
510 CALL CHAR(149,"00000000E0FFFF22")
520 CALL CHAR(150,"18183C5A5A18183C")
530 CALL CHAR(152,"FFC3A49999A542FF")
540 CALL CLEAR
550 CALL SCREEN(6)
560 CALL HCHAR(4,11,141)
570 CALL HCHAR(5,11,142)
580 CALL HCHAR(6,12,136)
590 CALL HCHAR(6,13,137)
600 CALL HCHAR(6,15,138)
610 CALL HCHAR(6,16,139)
620 CALL HCHAR(6,19,145)
630 CALL HCHAR(6,20,144)
640 CALL HCHAR(6,21,144)
650 CALL HCHAR(6,22,148)
660 CALL HCHAR(5,20,146)
670 CALL HCHAR(5,21,147)
680 CALL HCHAR(6,14,140)
690 CALL HCHAR(6,11,143)
700 CALL HCHAR(17,1,131)
710 CALL HCHAR(17,2,134)
720 CALL VCHAR(7,1,120,18)
730 FOR I=3 TO 10
740 CALL HCHAR(17,I,135)
750 NEXT I
760 CALL VCHAR(23,11,131,2)
770 CALL VCHAR(7,11,132,15)
780 CALL HCHAR(22,11,133)
790 CALL HCHAR(17,3,135,8)
800 FOR I=3 TO 9 STEP 2
810 CALL HCHAR(17,I,128)
820 NEXT I
830 CALL HCHAR(6,1,32,10)
840 CALL VCHAR(7,11,132,15)
850 CALL HCHAR(6,17,32,2)
860 CALL HCHAR(6,23,32,10)
870 CALL HCHAR(6,19,145)
880 CALL HCHAR(6,20,131,2)
890 CALL HCHAR(6,22,148)
900 FOR I=13 TO 27 STEP 2
910 CALL HCHAR(22,I,128)
920 NEXT I
930 FOR I=12 TO 26 STEP 2
940 CALL HCHAR(22,I,135)
950 NEXT I
960 CALL HCHAR(22,29,130)
970 CALL HCHAR(22,30,131,3)
980 FOR I=3 TO 9 STEP 2
990 CALL HCHAR(17,I,128)
1000 NEXT I
1005 REM POSITION DIFFERENT LAYERS OF ROCK
1010 RANDOMIZE
1020 C=12
1030 R=7
1040 A=INT(4*RND)+12
1050 CALL HCHAR(R,C,A*8,3)
1060 C=C+3
1070 IF C>32 THEN 1080 ELSE 1040
1080 C=12
1090 R=R+1
1100 IF R=22 THEN 1090 ELSE 1110
1110 IF R=25 THEN 1120 ELSE 1040

```

```

1120 C=2
1130 R=7
1140 A=INT(4*RND)+12
1150 CALL HCHAR(R,C,A*8,3)
1160 C=C+3
1170 IF C>10 THEN 1180 ELSE 1140
1180 C=2
1190 R=R+1
1200 IF R=17 THEN 1210 ELSE 1220
1210 R=R+1
1220 IF R=25 THEN 1230 ELSE 1140
1230 FOR I=1 TO 3

```

```

1240 C=INT(17*RND)+13
1250 CALL VCHAR(7,C,96,10)
1260 NEXT I
1270 FOR I=1 TO 3
1280 C=INT(9*RND)+2
1290 CALL VCHAR(8,C,96,7)
1300 NEXT I
1310 FOR I=1 TO 4
1320 R=INT(14*RND)+7
1330 C=INT(7*RND)+2
1340 CALL HCHAR(R,C,152,3)
1350 CALL VCHAR(R,C+2,152,2)
1360 NEXT I
1370 FOR I=1 TO 12
1380 R=INT(16*RND)+7
1390 C=INT(19*RND)+12
1400 CALL HCHAR(R,C,152,2)
1410 CALL VCHAR(R,C+1,152,3)
1420 NEXT I
1430 CALL HCHAR(22,28,129)
1440 CALL COLOR(13,2,15)
1450 CALL VCHAR(7,1,120,18)
1460 CALL HCHAR(17,2,134)
1470 CALL COLOR(14,15,6)
1480 CALL COLOR(15,2,6)
1485 REM EXPLOSION
1490 CALL SOUND(1000,-4,0)
1500 CALL SOUND(100,-7,0)

```

```

1510 CALL HCHAR(22,26,134)
1520 CALL SOUND(100,-6,0)
1530 CALL HCHAR(22,25,144)
1540 CALL SOUND(50,-7,2)
1550 CALL HCHAR(22,18,144,7)
1560 CALL SOUND(600,-6,2)
1570 CALL HCHAR(22,17,130)
1580 CALL SOUND(1000,-7,0)
1590 CALL HCHAR(22,15,134)
1600 CALL HCHAR(22,14,130)
1610 CALL HCHAR(22,11,134)
1620 FOR I=1 TO 3
1630 CALL HCHAR(6,2,150)
1640 FOR D=1 TO 200
1650 NEXT D
1660 CALL HCHAR(6,2,32)
1670 FOR D=1 TO 100
1680 NEXT D
1690 CALL SOUND(50,1000,5)
1700 NEXT I
1710 CALL HCHAR(6,2,150)
1715 REM SET UP START OF COUNT
1720 TIME=90
1730 R=6
1740 C=2
1750 TIME=TIME-1

```



```

1760 IF LEN(STR$(TIME))=1 THEN 1770 ELSE 1780
1770 CALL HCHAR(4,26,32)
1780 FOR I=1 TO LEN(STR$(TIME))
1790 CALL HCHAR(4,I+24,ASC(STR$(TIME),I,1))
1800 NEXT I
1810 CALL SOUND(100,-4,2)
1820 CALL KEY(0,K,S)
1825 REM TEST FOR 'OUT OF TIME'
1830 IF TIME=0 THEN 3380
1840 IF K=68 THEN 1900 ELSE 1850
1850 IF K=83 THEN 1920 ELSE 1860
1860 IF K=69 THEN 1940 ELSE 1870
1870 IF K<>88 THEN 1750 ELSE 1880

```


99/4A PROGRAM

```

1880 DR=+1
1890 GOTO 1950
1900 DC=+1
1910 GOTO 1990
1920 DC=-1
1930 GOTO 1990
1940 DR=-1
1950 RR=R
1960 CC=C
1970 R=R+DR
1980 GOTO 2020
1990 CC=C
2000 RR=R
2010 C=C+DC
2020 IF C>32 THEN 2030 ELSE 2050
2030 C=32
2040 GOTO 1750
2050 IF C<1 THEN 2060 ELSE 2080
2060 C=1
2070 GOTO 1750
2080 IF R<6 THEN 2090 ELSE 2110
2090 R=6
2100 GOTO 1750
2110 IF R>24 THEN 2120 ELSE 2140
2120 R=24
2130 GOTO 1750
2135 REM TEST FOR HARDNESS OF ROCK
2140 CALL HCHAR(R,C,X)
2150 IF X=96 THEN 2370
2160 IF X=104 THEN 2450
2170 IF X=112 THEN 2560
2180 IF X=120 THEN 2820
2190 IF X=32 THEN 2920
2200 IF X=131 THEN 2950
2210 IF X=128 THEN 2990
2220 IF X=134 THEN 3020
2230 IF X=144 THEN 3020
2240 IF X=130 THEN 3020
2250 IF X=148 THEN 3020

```

```

2260 IF X=145 THEN 3020
2270 IF X=135 THEN 2990
2280 IF X=129 THEN 3060
2290 IF X=143 THEN 3460
2300 IF X=132 THEN 3480
2310 IF X=136 THEN 3550
2320 IF X=137 THEN 3550
2330 IF X=138 THEN 3550
2340 IF X=139 THEN 3550
2350 IF X=152 THEN 2660
2360 GOTO 1750
2370 CALL COLOR(9,11,11)
2380 CALL SOUND(200,30000,30,30000,30,440,30,-4,0)
2390 CALL SOUND(400,30000,30)
2400 CALL COLOR(9,2,2)
2410 GOSUB 3220
2420 CALL HCHAR(RR,CC,135)
2430 CALL HCHAR(R,C,150)
2440 GOTO 1750
2450 REM
2460 FOR D=1 TO 2
2470 CALL COLOR(10,10,10)
2480 CALL SOUND(200,30000,30,440,12,440,30,-4,0)
2490 CALL SOUND(400,30000,30)
2500 CALL COLOR(10,2,2)
2510 GOSUB 3220
2520 NEXT D
2530 CALL HCHAR(RR,CC,135)
2540 CALL HCHAR(R,C,150)
2550 GOTO 1750
2560 FOR D=1 TO 3
2570 CALL COLOR(11,5,5)
2580 CALL SOUND(200,30000,30,880,12,440,30,-4,0)
2590 CALL SOUND(400,30000,30)
2600 CALL COLOR(11,2,2)
2610 GOSUB 3220
2620 NEXT D
2630 CALL HCHAR(RR,CC,135)
2640 CALL HCHAR(R,C,150)
2650 GOTO 1750
2660 FOR D=1 TO 3
2670 CALL COLOR(16,16,2)
2680 CALL SOUND(200,30000,30,1000,12,440,30,-4,0)
2690 CALL COLOR(16,2,2)
2700 NEXT D
2710 IF K=68 THEN 2760

```

```

2720 IF K=83 THEN 2780
2730 IF K=69 THEN 2800
2740 R=R-1
2750 GOTO 1750
2760 C=C-1
2770 GOTO 1750
2780 C=C+1
2790 GOTO 1750
2800 R=R+1
2810 GOTO 1750
2820 FOR D=1 TO 4
2830 CALL COLOR(12,4,4)
2840 CALL SOUND(150,30000,30,880,12,440,30,-4,0)
2850 CALL SOUND(400,30000,30)
2860 CALL COLOR(12,2,2)
2870 GOSUB 3220
2880 NEXT D
2890 CALL HCHAR(RR,CC,135)
2900 CALL HCHAR(R,C,150)
2910 GOTO 1750
2920 CALL HCHAR(RR,CC,32)
2930 CALL HCHAR(R,C,150)
2940 GOTO 1750
2950 CALL SOUND(100,200,4,210,4,-4,3)
2960 CALL HCHAR(RR,CC,135)
2970 CALL HCHAR(R,C,150)
2980 GOTO 1750
2990 CALL HCHAR(RR,CC,135)
3000 CALL HCHAR(R,C,150)
3010 GOTO 1750
3020 CALL SOUND(100,110,9,140,9,440,30,-4,0)
3030 CALL HCHAR(RR,CC,135)
3040 CALL HCHAR(R,C,150)
3050 GOTO 1750
3060 FOR I=1 TO 4
3070 CALL SOUND(100,1000,2)
3080 FOR D=1 TO 50
3090 NEXT D
3100 NEXT I
3105 REM WINNING ROUTINE
3110 M$="YOU DID IT!"
3120 R=18
3130 C=19
3140 GOSUB 3310
3150 M$="PRESS ANY KEY"
3160 R=20
3170 C=18
3180 GOSUB 3310
3190 CALL KEY(0,K,S)
3200 IF S=0 THEN 3190
3210 GOTO 780
3215 REM TIME COUNTDOWN ROUTINE
3220 TIME=TIME-1
3230 IF LEN(STR$(TIME))=1 THEN 3240 ELSE 3250
3240 CALL HCHAR(4,26,32)
3250 FOR I=1 TO LEN(STR$(TIME))
3260 CALL HCHAR(4,I+24,ASC(SEG$(STR$(TIME),I,1)))
3270 NEXT I
3280 CALL SOUND(100,-4,2)
3290 IF TIME=0 THEN 3380
3300 RETURN
3305 REM PRINT MESSAGE ROUTINE
3310 FOR I=1 TO LEN(M$)

```

```

3320 CD=ASC(SEG$(M$,I,1))
3330 CALL HCHAR(R,C,CD)
3340 C=C+1
3350 NEXT I
3360 RETURN
3370 STOP
3375 REM LOSE ROUTINE
3380 CALL SOUND(100,200,4)
3390 CALL SOUND(250,110,1)
3400 M$="***FAILED***"
3410 R=18
3420 C=17
3430 GOSUB 3310
3440 GOTO 3150
3450 END
3460 C=10
3470 GOTO 1750
3480 FOR D=1 TO 4
3490 CALL SOUND(200,400,4,200,4,-4,2)
3500 GOSUB 3220
3510 NEXT D
3520 CALL HCHAR(RR,CC,135)
3530 CALL HCHAR(R,C,150)
3540 GOTO 1750
3550 R=7
3560 GOTO 1750

```


How's your memory? It needs to be good to escape from the maze

How it works

40-100 define characters
120-170 set colours
200-370 lays out screen, prints score, GOSUB to walls lay out
380-400 print indicator showing level
430-460 wait for key press, check if maze finished
480-590 check E pressed, test for fruit ahead, move upwards and beep
610-720 as above, except key X, move down
740-840 key S pressed. Check for invisible door, moves through if there is GOSUBS score increment, or decrement if no door.
860-910 routine for printing on screen
930-1170 set flash time according to level draw walls and places random doors
1180-1340 randomise to see of to print fruit check if fruit all used if not, GOSUB print fruit.
1270-1340 Print fruit.
1360-1450 Effects for eating fruit increase score
1470-1580 flash openings in wall for set time
1600-1700 effects for moving through opening
1720-1830 same for failing to find opening correctly
1850-1980 game finished, check score
2000-2150 game over, replay opportunity
2170-2260 new maze, score good enough
2280-2340 variables set-up

In this game, you have to guide a figure through a maze — but it's not as easy as it sounds!

A number of vertical walls are printed down the screen, with channels in between. Random doors are set into each wall.

These openings in the walls flash for a set period of time at the beginning of the game, during which you must memorise their locations.

After 15 seconds (on the easiest level) the doors close up. You must now guide your figure across the screen using keys S and X to move up and down and E to try to move through the now invisible holes in the walls.

The figure leaves a trail behind him. For each door you

You've got just 15 seconds to memorise the location of the doors in John Powers's game for the standard TI-99/4A. And time gets shorter as your memory improves

Hints on conversion

If you would like to try to convert this program to another machine, remember that the Texas has a 24 by 32 screen, and a good amount of space can be saved by using multiple line statements — the program only takes up 5½K anyway — not available on the standard TI.

Here are the more unusual commands:

CALL KEY(M,K,S) Where K is the ASCII value of the key pressed, and S returns 1 if pressed. You can replace this with INKEY or GET.

CALL SOUND(D,N,V) Where D is duration, N is note in Hertz, V is the volume. Most machines have some sound capabilities — so it's easier to write your own.

CALL CHAR(A,NS) Defines characters in Hexadecimal. N is the ASCII character to be redefined. Characters are defined left-to-right from the top. Each digit represents the pattern in a 1×4 horizontal block of pixels; so there are 2 Hex digits for each line of the 8×8 square.

CALL HCHAR(R,C,A,P) Positions ASCII character A, and repeats horizontally from coordinates C and R P times.

CALL VCHAR(R,C,A,P) As above except repeats vertically.

CALL COLOUR(S,F,B) Defines character foreground and background colour. Replace with GCOL or INK or other command.

CALL SCREEN(X) Sets screen colour. Replace with PAPER or VDU 19 or COLOUR statement.

CALL CLEAR Clears screen. Use CLS.

CALL GCHAR(R,C,V) Returns ASCII code of character at position R,C in variable V. Use a POINT or screen PEEK.

successfully pass through, your score increases by 100. If you try to pass through where there is no door you lose 35 points.

When you eat a fruit (placed randomly in the maze), the doors become visible for five seconds and you score bonus points.

There are three levels of skill. An indicator at top left

Variables

Z row of print position
V columns of print position
G level
CL column of figure
RW row of figure
GC ASCII code from GCHAR
CT column position of walls
RA/RB position of doors in walls
COUNT number of fruit on screen
BF number of fruit available
FI random position of fruit
SC score
MS screen messages

shows you current level, you pass onto the next level automatically if your score is high enough. The game is fairly easy to play at the lowest level but it takes a good deal of skill to navigate your way around the harder mazes.

```
30 REM **DEFINE CHARACTERS**
40 CALL CHAR(96,"0000001818000000")
50 CALL CHAR(104,"030404307E7E7E30")
60 CALL CHAR(112,"000000FF00FF0000")
70 CALL CHAR(120,"1818181818181818")
80 CALL CHAR(129,"1818181818181818")
90 CALL CHAR(136,"FFFFFFFFFFFFFFFF")
100 CALL CHAR(145,"18307EFF7E301800")
110 REM **SET COLOURS**
120 CALL COLOR(9,2,8)
130 CALL COLOR(10,13,8)
140 CALL COLOR(11,10,15)
150 CALL COLOR(13,5,8)
160 CALL COLOR(14,8,15)
170 CALL COLOR(15,7,8)
180 GOSUB 2270
190 REM **SET UP SCREEN**
200 CALL CLEAR
210 CALL SCREEN(15)
220 CALL HCHAR(5,1,136,576)
230 MS="MEMORY MAZE"
240 Z=3
250 V=2
260 GOSUB 870
270 MS="SCORE="&STR$(SC)
280 Z=3
290 V=20
300 GOSUB 870
310 CALL HCHAR(4,3,112,28)
320 CALL HCHAR(2,3,112,28)
330 CALL HCHAR(23,3,112,28)
340 GOSUB 920
350 MS="
```


TI-99/4A PROGRAM

```

360 Z=24
370 GOSUB 850
380 FOR LS=1 TO G
390 CALL HCHAR(1,2+LS,145)
400 NEXT LS
410 CALL HCHAR(RW,CL,145)
420 REM **SCAN KEYBOARD**
430 CALL KEY(0,K,S)
440 IF S=0 THEN 430
450 IF CL<=3 THEN 460 ELSE 480
460 GOTO 1840
470 REM **KEY E**
480 IF K<>69 THEN 610
490 CALL HCHAR(RW,CL,96)
500 IF RW<=5 THEN 510 ELSE 520
510 RW=23
520 RW=RW-1
530 CALL GCHAR(RW,CL,GC)
540 IF GC=104 THEN 550 ELSE 570
550 GOSUB 1350
560 GOTO 430
570 CALL HCHAR(RW,CL,145)
580 CALL SOUND(-50,-1,2)
590 GOTO 430

```

```

600 REM **KEY X**
610 IF K<>88 THEN 740
620 CALL HCHAR(RW,CL,96)
630 IF RW>=22 THEN 640 ELSE 650
640 RW=4
650 RW=RW+1
660 CALL GCHAR(RW,CL,GC)
670 IF GC=104 THEN 680 ELSE 700
680 GOSUB 1350
690 GOTO 430
700 CALL HCHAR(RW,CL,145)
710 CALL SOUND(-50,-1,2)
720 GOTO 430
730 REM **KEY S**
740 IF K<>83 THEN 430
750 CALL GCHAR(RW,CL-1,GC)
760 IF GC=120 THEN 790 ELSE 770
770 GOSUB 1720
780 GOTO 430
790 CALL HCHAR(RW,CL-1,96,2)
800 CL=CL-2
810 CALL HCHAR(RW,CL,145)
820 GOSUB 1600
830 IF CL<=3 THEN 1840
840 GOTO 430
850 REM **PRINT ROUTINE**
860 V=16-INT(LEN(M$)/2)
870 FOR I=1 TO LEN(M$)
880 CD=ASC(SEG$(M$,I,1))
890 CALL HCHAR(Z,V+I,CD)
900 NEXT I
910 RETURN
920 REM **SET UP WALLS**
930 CALL COLOR(12,2,8)
940 IF G=1 THEN 950 ELSE 970
950 M$="15 SECS TO MEMORIZE THE MAZE."
960 GOTO 980
970 M$="8 SECS TO MEMORIZE THE MAZE."
980 Z=24

```

```

990 GOSUB 850
1000 FOR CT=30 TO 4 STEP -2
1010 CALL VCHAR(5,CT,129,18)
1020 NEXT CT
1030 FOR CT=30 TO 4 STEP -2
1040 RANDOMIZE
1050 RA=INT((21-5+1)*RND)+5
1060 RB=INT((21-5+1)*RND)+5
1070 IF G=1 THEN 1080 ELSE 1110
1080 CALL VCHAR(RA,CT,120)
1090 CALL VCHAR(RB,CT,120)
1100 GOTO 1180
1110 IF G=2 THEN 1120 ELSE 1170
1120 IF RA=21 THEN 1040
1130 CALL VCHAR(RA,CT,120)
1140 RB=RA+1
1150 CALL VCHAR(RB,CT,120)
1160 GOTO 1180
1170 CALL VCHAR(RA,CT,120)
1180 IF INT(RND*7)+1=3 THEN 1190 ELSE 1220
1190 IF COUNT=6F THEN 1220 ELSE 1200

```

```

1200 COUNT=COUNT+1
1210 GOSUB 1280
1220 NEXT CT
1230 GOSUB 1470
1240 IF G=1 THEN 1250 ELSE 1260
1250 GOSUB 1470
1260 RETURN
1270 REM **PRINT FRUIT**
1280 RANDOMIZE
1290 F1=INT((21-5+1)*RND)+5
1300 IF F1=RA THEN 1280
1310 IF F1=RB THEN 1280
1320 IF CT<=4 THEN 1340
1330 CALL HCHAR(F1,CT-1,104)
1340 RETURN
1350 REM **EAT FRUIT**
1360 CALL SOUND(-500,-5,2)
1370 CALL HCHAR(RW,CL,96)
1380 SC=SC+200
1390 M$=STR$(SC)&" "
1400 Z=3
1410 V=26
1420 GOSUB 870
1430 CALL HCHAR(RW,CL,145)
1440 GOSUB 1470
1450 RETURN
1460 REM **FLASH WALLS**
1470 FOR X=1 TO 5
1480 CALL COLOR(12,2,8)
1490 CALL SOUND(-100,880,2)
1500 FOR T=1 TO 250
1510 NEXT T
1520 CALL COLOR(12,8,8)
1530 CALL SOUND(-100,880,2)
1540 FOR T=1 TO 250
1550 NEXT T
1560 NEXT X
1570 CALL COLOR(12,5,8)
1580 RETURN
1590 REM **THROUGH WALL**

```


TI-99/4A PROGRAM

```

1600 CALL SOUND(-500,440,2)
1610 CALL SOUND(-500,523,2)
1620 CALL SOUND(-500,659,2)
1630 CALL SOUND(-500,440,2)
1640 CALL SOUND(-500,784,2)
1650 SC=SC+100
1660 M$=STR$(SC)
1670 Z=3
1680 V=26
1690 GOSUB 870
1700 RETURN
1710 REM **HIT WALL**
1720 CALL SOUND(-500,196,2)
1730 CALL SOUND(-500,165,2)
1740 CALL SOUND(-500,131,2)
1750 IF SC<=0 THEN 1830
1760 SC=SC-35
1770 IF SC<0 THEN 1780 ELSE 1790
1780 SC=0
1790 M$=STR$(SC)&" "

```

```

2190 RW=14
2200 CL=31
2210 SC=0
2220 COUNT=0
2230 IF G>=3 THEN 2250
2240 BF=BF-1
2250 G=G+1
2260 GOTO 190
2270 REM **SET-UP VARIABLES**
2280 COUNT=0
2290 BF=3
2300 SC=0
2310 G=1
2320 RW=14
2330 CL=31
2340 RETURN

```



```

1800 Z=3
1810 V=26
1820 GOSUB 870
1830 RETURN
1840 REM **FINISH ROUTINE**
1850 CALL SOUND(-500,523,2)
1860 CALL SOUND(-500,659,2)
1870 CALL SOUND(-500,784,2)
1880 CALL SOUND(-500,659,2)
1890 CALL SOUND(-500,880,2)
1900 CALL SOUND(-500,698,2)
1910 CALL SOUND(-750,698,2,880,2,1047,2)
1920 M$="WELL DONE YOU MADE IT!"
1930 Z=24
1940 GOSUB 850
1950 IF (G=1)*(SC<1000) THEN 1990
1960 IF (G=2)*(SC<900) THEN 1990
1970 IF (G>=3)*(SC<700) THEN 1990
1980 GOTO 2160
1990 REM **GAME OVER**
2000 CALL CLEAR
2010 CALL SCREEN(8)
2020 PRINT "*****"
2030 PRINT "*****GAME OVER*****"
2040 PRINT "*****"
2050 PRINT "PRESS.."
2060 PRINT "*****"
2070 PRINT "*****[SPACE BAR]*****"
2080 PRINT "*****"
2090 PRINT " ..TO PLAY AGAIN."
2100 CALL KEY(0,K,S)
2110 IF S=0 THEN 2100
2120 CALL CLEAR
2130 IF K=32 THEN 2150
2140 END
2150 GOTO 180
2160 REM **NEW SHEET**
2170 FOR X=1 TO 500
2180 NEXT X

```

STARTTECH

Absolutely
the lowest
prices!



BLANK DATA CASSETTES

£5.95

Inc VAT
Postage and
Packing 45p

Computer Grade Data Cassettes suitable for use in all of today's home computers.
Each comes with its own case.

Tick for further information
☐ VIC 20 PRODUCTS
☐ COMMODORE 64

All prices subject to availability or change without notice.

PLEASE SEND ME

Pack of 10 Data Cassettes

Qty Price TOTAL

£6.40 inc VAT

Name

Address

I enclose Cheque, P/O for

CREDIT CARD ☐ ACCESS ☐ BARCLAY CARD ☐ (24hr. ANSWERPHONE SERVICE)

HCW18

STARTTECH 208 Aigburth Rd, Aigburth, Liverpool L17 051 727-7267

Can you save Miss Muffet from the spider?

Hints on conversion

The program should be convertible for use on any computer with user-defined characters.

CALL CHAR (ASCII code number, hexadecimal string) Defines characters used in the program. The hexadecimal string describes the pattern to be given to the character with ASCII code stated.

CALL HCHAR (row number, column number, ASCII code, number of repetitions) Places character on screen at row and column specified, and repeats it horizontally the number of times stated. If this fourth value is omitted then the character is displayed only once. The character used is the one whose ASCII code appears in the command.

CALL VCHAR (row number, column number, ASCII code, number of repetitions) Works in the same way as CALL HCHAR only repeats vertically instead of horizontally.

CALL GCHAR (row number, column number, numeric, numeric variable) Equivalent of PEEK, assigns ASCII code of character at row and column stated to numeric variable.

CALL CLEAR Clears the screen

CALL SCREEN (Colour code) Specifies screen colour, using colour code.

CALL COLOR (character set number, foreground colour code, background colour code) Specifies foreground and background colours for all characters in set stated.

CALL SOUND (duration, frequency, volume) Produces sound with duration, frequency, and volume specified. Duration is in milliseconds, frequency in Hertz (or noise code), and volume in scale from 0 loudest to 30 quietest.

CALL KEY (key unit, k,s) Equivalent of INKEYS in other computers, returns code if key pressed in variable k.

Poor Miss Muffet. She's been chased by a spider into a labyrinth. It's up to you to guide her to the golden door.
Lance Booth wrote this program for the standard TI-99/4A

In this game of luck, skill and judgement, an evil spider has chased Miss Muffet into a labyrinth, which is about to fill with deadly green slime. Can you help her to escape in time?

You must guide her — using the four arrow keys with the alpha lock button down — through the labyrinth to the golden door, the only way out.

The spider is patrolling the passages of the labyrinth, spinning webs to block Miss Muffet's

escape. Don't let the spider catch Miss Muffet or it will devour her.

Take care not to let her get caught in a web or to touch the walls of the labyrinth which are covered with the spider's venom.

You set your own difficulty

level, when line 330 asks for your time limit. This sets the limit at which the program leaves the game loop from line 690, and moves to the out of time routine at line 930. A time limit of 150 could be quite a challenge to start with, even if you get the right breaks.

Once you've had some practice, trying to work down to a limit of 100 becomes an addiction.

A variation on the game, where the spider can be made to move from bottom to top, instead of top to bottom, can be achieved by changing line 490 from FOR SR = 3 TO 23 STEP 2 to FOR SR = 23 TO 3 STEP -2.

How it works

130-250 define characters and colours

260 sets screen colour to black

270-310 assign initial value to variables

320 clears screen

330 asks for time limit

340 clears screen ready for game

350-450 display labyrinth

460 displays Miss Muffet

470 displays Golden Door

480-780 main game loop

490-770 count down spider movement rows

500-510 & 750 control spider's movement along columns

520 checks keyboard

530 clears old player from screen

540-610 check for movement

620-660 check new screen positions

670 displays new player

680 adds one to penalty score

690 tests to see if limit is exceeded

700-710 check new spider position

720-740 move spider

760 displays web

790-810 touched the wall

820-890 eaten by spider

900-920 trapped in a web

930-990 ran out of time

1000-1060 play music

1070-1220 escape

1230-1250 display message

1260-1280 want to play again?

```

110 REM BY LANCE BOOTH
120 REM SET CHARACTERS
130 CALL CHAR(41,"FFFFFFFFFFFFFFFF")
140 CALL CHAR(43,"183C7E183C7E1818")
150 CALL CHAR(128,"183C7EFFFFFFFF")
160 CALL CHAR(136,"7E5A7E7EA5A5A5")
170 CALL CHAR(152,"18187E183C7E2424")
180 REM SET COLOURS
190 CALL COLOR(13,12,1)
200 CALL COLOR(14,15,1)
210 CALL COLOR(15,10,10)
220 CALL COLOR(16,11,1)
230 FOR I=2 TO 8
240 CALL COLOR(I,3,1)
250 NEXT I
260 CALL SCREEN(2)
270 MR=23
280 MC=16
290 SCORE=0
300 RANDOMIZE
310 DC=INT(29*RND)+2
320 CALL CLEAR
330 INPUT "WHAT IS YOUR TIME LIMIT?"
    :LIMIT
    
```



TI-99/4A PROGRAM

```

340 CALL CLEAR
350 REM SET SCREEN
360 CALL HCHAR(2,1,144,32)
370 CALL HCHAR(24,1,144,32)
380 CALL VCHAR(1,1,144,24)
390 CALL VCHAR(1,32,144,24)
400 CALL HCHAR(1,1,43,32)
410 FOR ROW=4 TO 22 STEP 2
420 CALL HCHAR(ROW,1,144,32)
430 X=INT(29*RND)+2
435 Z=INT(29*RND)+2
440 CALL HCHAR(ROW,X,32)
445 CALL HCHAR(ROW,Z,32)

```

```

450 NEXT ROW
460 CALL HCHAR(MR,MC,152)
470 CALL HCHAR(2,DC,128)
480 REM GAME LOOP
490 FOR SR=3 TO 23 STEP 2
500 Y=INT(29*RND)+2
510 FOR SC=2 TO Y
520 CALL KEY(3,K,S)
530 CALL HCHAR(MR,MC,32)
540 IF K<>83 THEN 560
550 MC=MC-1
560 IF K<>68 THEN 580
570 MC=MC+1
580 IF K<>69 THEN 600

```

```

590 MR=MR-1
600 IF K<>88 THEN 620
610 MR=MR+1
620 CALL GCHAR(MR,MC,GET)
630 IF GET=144 THEN 790
640 IF GET=136 THEN 820
650 IF GET=128 THEN 1070
660 IF GET=42 THEN 900
670 CALL HCHAR(MR,MC,152)
680 SCORE=SCORE+1
690 IF SCORE>LIMIT THEN 930
700 CALL GCHAR(SR,SC,GET)
710 IF GET=152 THEN 820
720 CALL HCHAR(SR,SC,136)
730 CALL SOUND(100,110,0)
740 CALL HCHAR(SR,SC,32)

```

```

750 NEXT SC
760 CALL HCHAR(SR,SC,42)
770 NEXT SR
780 GOTO 480
790 GOSUB 1000
800 M$="MISS MUFFET TOUCHED THE WALL"
810 GOTO 1230
820 FOR I=1 TO 10
830 CALL HCHAR(MR,MC,152)
840 CALL SOUND(50,-5,0)
850 CALL HCHAR(MR,MC,136)
860 NEXT I
870 GOSUB 1000
880 M$="THE SPIDER ATE MISS MUFFET"
890 GOTO 1230
900 GOSUB 1000
910 M$="MISS MUFFET'S TRAPPED IN A WEB"
920 GOTO 1230
930 FOR R=23 TO 3 STEP -2
940 CALL HCHAR(R,2,41,30)
950 CALL SOUND(-500,110+(R*10),0)
960 NEXT R
970 GOSUB 1000
980 M$="THE SLIME KILLED MISS MUFFET"
990 GOTO 1230
1000 REM DEATH TUNE
1010 TONE=1100
1020 FOR C=1 TO 10
1030 CALL SOUND(+500,TONE,0)
1040 TONE=TONE-110
1050 NEXT C

```

```

1060 RETURN
1070 NOTE=110
1080 FOR C=1 TO 10
1090 CALL SOUND(-500,NOTE,0)
1100 NOTE=NOTE+110
1110 NEXT C
1120 CALL HCHAR(1,MC,152)
1130 M$="WELL DONE MISS MUFFET ESCAPED"
1140 CALL CLEAR
1150 PRINT "YOUR SCORE IS";SCORE
1160 PRINT
1170 PRINT "PENALTY POINTS"
1180 PRINT
1190 CALL HCHAR(9,1,152,32)
1200 CALL HCHAR(10,1,43,32)
1210 CALL HCHAR(12,1,42,32)
1220 CALL HCHAR(13,1,136,32)
1230 FOR J=1 TO LEN(M$)
1240 CALL HCHAR(11,1+J,ASC(SEG$(M$,J,1)))
1250 NEXT J
1260 INPUT "WANT TO PLAY AGAIN? ";A$
1270 IF SEG$(A$,1,1)="Y" THEN 270
1280 IF SEG$(A$,1,1)<>"N" THEN 1260
1290 END

```


Computer flights of fantasy

Know the fear of flying with these aerial adventures for a range of micros.

Nightflight Dragon 32 £7.95

Salamander Software, 27 Ditching Rise, Brighton, East Sussex BN1 4QL

If this is a flight simulator I am green with yellow stripes (I'm not!). The Dragon can do, and has done, much better stuff than this. The best thing about the game is the opening title page, and even then they played Those Magnificent Men in their Flying Machines wrong!

Low resolution graphics, bad use of colour and unrealistic

sound. The screen did not look at all like a plane cockpit...no dials, all the same presentation.

I would now like to say "on the plus side", but there isn't a plus side. I suppose the instructions were clear(ish) but they weren't relevant to the game.

Psion's flight simulation program for the ZX81 is much better than this, it has higher resolution graphics, is more responsive and more realistic.

instructions
playability
graphics
value for money

P.S.

80%
10%
10%
3%



Night Flight TI-99/4A £5.00

Stainless Software, 10 Alstone Road, Stockport, Cheshire

As commander of the Starship Patscram, it is your task to clear a landing strip in a hostile city defended by ground lasers and spacecraft.

This is an addictive game which requires dexterity to manipulate the 10 keys needed to play fully. There are plenty of skill levels and great scope for player improvement.

You score points for forward movement and firing splat bombs but these diminish your fuel. Points are also awarded for

destroying the enemy, safe landing and refuelling. High scores can be saved on tape.

This is a lengthy program to load but there is only a minimum delay in setting up. It has at least two bugs:

- 1 The acceleration command lets your ship eat the right-hand boundary.
- 2 Bomb speed keys are inoperable if ship's control keys are reselected.

But these are minor irritations in what is otherwise a very good TI game.

instructions
playability
graphics
value for money

C.E.

90%
80%
90%
90%



Winging It TI-99/4A £10

Not Polyoptics, from Stainless Software, 10 Alstone Road, Stockport, Cheshire SK4 5AH

Could this be the first flight simulator for the TI? Once you've mastered basic flying, you can progress to one of three games. In game 1 you locate and fly over a meteorite, game 2 sets you the task of finding and destroying a flying saucer, and in game 3 you are pitted against an enemy plane.

There's no 3D view of the runway or the like, but there is a first-class display of the instru-

ment panel, and realistic engine noises. Comprehensive instructions are supplied.

The instructions warn that beginners may have a rough ride, and they aren't kidding. Taking off was OK, but once in the air I found an affinity for flying upside down and making unscheduled and rather violent landings.

Still, practice makes perfect. I thought this was an excellent program, which should appeal to budding pilots of all ages.

instructions
playability
graphics
value for money

D.B.
100%
90%
90%
90%



Landfall BBC B £7.95

Virgin Games, 61/13 Portobello Road, London W11

This is a sort of sophisticated Lunar Lander, though as the landing process is all done by instruments, it could be more aptly described as a spaceship landing simulator.

The overall effect is very convincing, colour is unobtrusive but effective, and the sounds (which you can switch off) are realistic.

Landing safely is no easy feat - you really have to watch all the gauges and keep a balance

to avoid being burned up on entry. I never managed to land at all, and would have liked some sort of variable difficulty level or practice option.

The program loads in three blocks, and after loading the first I had all the signs of a crash on my hands. After several attempts I just let it continue, and it loaded the second block - a screen picture, then the game.

instructions
playability
graphics
value for money

R.E.

80%
67%
73%
86%



Up Up and Away Atari 400/800 £14.95

Starcade, 15 Moorfields, Liverpool, L22 BQ

If you've worn your nerve-endings to a frazzle with pacman, invading creatures and perilous adventures, perhaps you think hot-air ballooning would make a nice quiet change. Take it from me - Up Up and Away will make you want to keep your feet firmly on the ground.

You take off with a limited amount of butane and ballast, and your aim is to fly round a cir-

cuit avoiding hazards on the way. These range from nasty little boys with slingshots, updraughts and downdraughts, thunderstorms, the local Flying School for the Shortsighted, and even incontinent pigeons!

The graphics are among the best I've ever seen, and the five levels of difficulty will provide a long-lasting challenge.

instructions
playability
graphics
value for money

J.P.

80%
85%
95%
85%



Take on your TI

Looking for a TI game? Our reviewers look at a selection for TI BASIC and Extended BASIC

Hang Glider Pilot £13 Extended BASIC

By Maple Leaf Micro Ware, from Stainless Software, 10 Alstone Road, Stockport, Cheshire SK4 5AH

If you have ever fancied soaring from a sheer cliff aboard a hang glider, but without the added thrill of breaking a leg or two, then this hang glider trainer is for you.

The object of the game, for one to four players, is to jump from a ridge and glide for the greatest distance and time before

landing safely in a designated landing zone. More difficult in practice than theory, a survival rate of around 30 per cent indicates (I hope) that the transfer is more difficult than the real thing.

You are presented with a graphical display of your position on the landscape together with 'instrument' readouts covering such factors as height, orientation, wind speed and direction. Clouds of different sizes on the display indicate the positions of thermals which give lift.

instructions
playability
graphics
value for money

V.A.
80%
80%
70%
40%



Devil Craze £12 Extended BASIC

By Maple Leaf Micro Ware, from Stainless Software, 10 Alstone Road, Stockport, Cheshire SK4 2DE

Your unlikely sounding hosts for this game are The Crazy Red Devil, The Great Imposter and The Five Big Bad Boys.

Three rules govern this game of quick reflexes and decision making. Rule one — if the centre box colour matches the left side, press 1 on the keyboard. If the centre box matches the right side, press 0. Rule two — if four red

devils appear, do the opposite, and finally rule three — ignore the great imposter.

Confused? Upon completion you are rated according to your score, descriptions ranging from "Genuine Wimp" to "Grand Master"!

At £12, plus an Extended BASIC cartridge, this simple game is likely to prove too expensive for most people. Good graphics and sound, superb presentation. Shame about the

instructions
playability
graphics
value for money

V.A.
70%
40%
80%
30%



Cut-off £5

Stainless Software, 10 Alstone Road, Stockport, Cheshire SK4 5AH

This is a fun game which one or two people can play on the standard machine (or faster with Extended BASIC) using keyboard or joysticks.

The object of the game is survival. As you go along you have to avoid houses and your own or opponents track. At first all obstacles are visible but as the game progresses they become invisible. Further on, a bouncing ball comes into play and also has to be avoided.

You gain points for track

laid and passing through flashing power squares. You have 11 lives so the game lasts a reasonable length of time.

Unfortunately the game surround flashes along with the power squares, which makes it hard on your eyes, and although the invisible obstacles can be seen momentarily when you pass through flashing squares they aren't revealed when you crash with them.

instructions
playability
graphics
value for money

C.E.

90%
75%
70%
70%



Splodge £5

Stainless Software, 10 Alstone Road, Stockport, Cheshire SK4 5AH

Cross the Gulf of Splodge, if you can! After listening to a repetitive tune for a full 1½ minutes the game was ready to play. I lost three lives in less than the set up time and the game was over. My second game was brought to an abrupt end by a 'bad subscript' report.

On subsequent attempts I managed to cross the gulf, dodging 10 rows of obstacles which moved from side to side. My task was made easier after every suc-

cessful crossing as some of the obstacles unplotted leaving a clear gap. Whether this is part of the program or an error I cannot say. At the end of the game the score, if high enough, is slotted into a 'score table' and the player is invited to play again.

The graphics are fair but a better choice of colour would enable the player to see his piece more easily. The over-use of sound slows the game down. This game is unrewarding —

instructions
playability
graphics
value for money

TI-99

C.E.

5%
30%
40%
30%



Tickworld £9

By Not Polyoptics, from Stainless Software, 10 Alstone Road, Stockport, Cheshire SK4 5AH

Tickworld takes place in a sparse jungle inhabited by eight man-sized and hungry ticks. You control a small figure, with the aid of the arrow keys on the keyboard, attempting to both avoid ending up as a mid-morning snack and to capture the eight ticks with the aid of an unlimited supply of nets and eight cages.

The only way to capture these nightmare-escapees is to throw nets around and hope they stumble into one as they move

relentlessly towards you. Luckily their brains stayed normal size and this is not difficult.

These are three skill levels to choose from, the only apparent difference being less trees to hide behind on the higher levels. Graphics and sound effects were quite impressive, adding much to the playability of the game.

My only criticisms are the slow speed of the game and the extreme difficulty in gaining a foothold against overwhelming odds.

instructions
playability
graphics
value for money

V.A.

50%
50%
80%
40%



Page 29

Ughh! Don't let the slugs get their revenge

Your terraced garden is infested with 20 giant slugs, all sucking the oxygen from the air.

To reach your seed-box and plant the seeds you must clear the slugs from each level by dropping slug pellets, using the space bar.

To reverse direction use the R key, but remember that the effort will reduce your oxygen reserve. Do not run out of pellets or descend a ladder before the slugs are killed or they will get their revenge.

Main variables

R row of man on screen
C column of man on screen
CE erase column of man
RE erase row of man
X define direction of man
SP quantity of slug pellets
SG number of slugs
L oxygen level

How it works

150-340 define characters and colours
350-420 input and check number of slug pellets
430-570 set up screen graphics
580-640 set initial variables and man's position
660-1200 set man moving, detect key presses and hits on slugs, descend ladders
1220-1760 lose and win routines, request another game
1780-1810 subroutine to clear screen

```
380 REM SET SCREEN
390 CALL CLEAR
400 CALL SCREEN(2)
410 CALL COLOR(1,11,11)
420 RANDOMIZE
430 FOR I=4 TO 24 STEP 5
440 CALL HCHAR(I,1,144,32)
450 NEXT I
460 CALL VCHAR(1,3,98,24)
470 CALL VCHAR(1,32,99,24)
480 CALL HCHAR(23,30,128)
490 CALL HCHAR(1,4,136,27)
500 FOR I=8 TO 23 STEP 5
510 FOR J=1 TO 5
520 SC=INT(27*RND)+4
530 CALL GCHAR(I,SC,SS)
540 IF SS<>32 THEN 520
550 CALL HCHAR(I,SC,104)
560 NEXT J
570 NEXT I
580 X=1
590 L=32
600 RE=1
610 CE=1
620 SG=5
630 R=3
```

Your garden turns into a battleground as you try to defeat 20 giant slugs in Barbara Baldwin's program for the TI-99/4A.
Will you risk typing it in?

```
100 REM SLUGS AND LADDERS
110 REM
120 REM BY B. BALDWIN
130 REM DEF GRAPHICS
140 CALL CLEAR
150 CALL CHAR(104,"000000007EFFFFFF")
160 CALL CHAR(100,"00000000C9C9FFFF")
170 CALL CHAR(96,"1C1C08FEDD1C1436")
180 CALL CHAR(136,"FFFFFFFF")
190 CALL CHAR(120,"000000383838")
200 CALL CHAR(128,"AA55AA55AA55AA55")
210 CALL CHAR(97,"9D9D89FF9D9D95B7")
220 CALL CHAR(99,"FFFFFFFFFFFFFFFF")
230 CALL CHAR(98,"81FF81FF81FF81FF")
240 CALL CHAR(144,"00FFFFFFFFFFFF00")
250 CALL CHAR(152,"10925438FE385492")
260 CALL CHAR(105,"10101092D6D67C38")
270 CALL COLOR(14,10,11)
280 CALL COLOR(16,14,11)
290 CALL COLOR(10,13,11)
300 CALL COLOR(9,2,11)
310 CALL COLOR(12,5,11)
320 CALL COLOR(13,9,12)
330 CALL COLOR(9,2,11)
340 CALL COLOR(15,2,16)
350 PRINT " HOW MANY SLUG PELLETS?":
360 INPUT " (20 TO 40) ":SP
370 IF (SP<20)+(SP>40) THEN 360
```

```
640 C=4
650 REM START GAME
660 CALL HCHAR(R,C,96)
670 IF SP=0 THEN 1320
680 CALL KEY(0,K,S)
690 RE=R
700 CE=C
710 IF S<>0 THEN 940
720 C=C+X
730 IF C>30 THEN 770
740 IF C=3 THEN 810
750 CALL HCHAR(RE,CE,32)
760 GOTO 660
770 CALL HCHAR(RE,CE,32)
780 IF (C=31)*(R=23) THEN 1420
790 C=31
800 GOTO 660
810 CALL HCHAR(RE,CE,32)
820 FOR I=R+1 TO R+5
830 CALL HCHAR(I,C,97)
840 CALL SOUND(100,200,1)
850 CALL HCHAR(I,C,98)
860 NEXT I
870 R=R+5
880 CALL HCHAR(R,3,98)
```

Hints on conversion
Texas BASIC has some unusual commands for graphics and sound:

CALL CHAR (number, hexadecimal string) controls the user definition of characters. Number refers to the ASCII code of the re-defined character.

CALL HCHAR (row, column, number) prints the character whose ASCII code is number at position row, column on the screen. The TI99/4A has a screen size of 24 rows by 32 columns.

CALL GCHAR (row, column, variable) is the equivalent of PEEK. The ASCII code of the character at position row, column on the screen is left in the designated variable.

CALL SOUND (d, f, v) sounds a note of frequency f, duration d milli-seconds and volume v.

CALL KEY (O, K, S), the equivalent of INKEYS, will leave the ASCII code of the key pressed in the variable K.

CALL CLEAR clears the screen. A direct equivalent of CLS.

```
890 X=1
900 IF SG>0 THEN 1230
910 SG=5
920 GOTO 640
930 REM DETECT KEYS
940 IF K<>82 THEN 1010
950 L=L-1
960 IF L=3 THEN 1710
970 CALL HCHAR(1,L,32)
980 X=-X
990 CALL SOUND(20,1600,5)
1000 GOTO 720
1010 IF K<>32 THEN 720
1020 SP=SP-1
1030 FOR I=R+2 TO R+5
1040 CALL GCHAR(I,C,HS)
1050 CALL HCHAR(I,C,120)
1060 CALL SOUND(20,400,1)
1070 CALL HCHAR(I,C,32)
1080 NEXT I
1090 IF HS<>128 THEN 1110
1100 CALL HCHAR(23,30,128)
1110 IF HS<>104 THEN 720
1120 REM HIT SLUG
1130 CALL HCHAR(R+5,C,104)
1140 CALL COLOR(10,14,11)
```


TI-99/4A PROGRAM

```

1150 SG=SG-1
1160 FOR I=5 TO 0 STEP -1
1170 CALL SOUND(100,-6,I)
1180 NEXT I
1190 CALL HCHAR(R+5,C,32)
1200 CALL COLOR(10,13,11)
1210 GOTO 720
1220 REM LOSE ROUTINES
1230 CALL HCHAR(R,4,96)
1240 FOR I=16 TO 1 STEP -1
1250 CALL HCHAR(R,I+3,104)
1260 CALL SOUND(100,-6,I)
1270 NEXT I
1280 GOSUB 1780
1290 PRINT " THE SLUGS SLAUGHTERED YOU":::
1300 GOTO 1670
1310 END
1320 FOR I=4 TO 31
1330 CALL COLOR(10,14,11)
1340 CALL HCHAR(R,I,104)
1350 CALL SOUND(50,-6,1)
1360 CALL COLOR(10,13,11)
1370 NEXT I
1380 GOSUB 1780
1390 PRINT " YOU RAN OUT OF PELLETS":::
1400 GOTO 1290
1410 REM WIN ROUTINE
1420 FOR I=30 TO 3 STEP -2
1430 CALL HCHAR(R,I,96)
1440 CALL SOUND(10,800,2)
1450 CALL HCHAR(R,I,32)
1460 NEXT I
1470 FOR I=23 TO 3 STEP -2
1480 CALL HCHAR(I,3,97)

```

```

1490 CALL SOUND(50,200,2)
1500 CALL HCHAR(I,3,98)
1510 NEXT I
1520 CALL HCHAR(3,2,96)
1530 FOR I=28 TO 5 STEP -4
1540 CALL HCHAR(23,I,105)
1550 CALL SOUND(20,1600,2)
1560 NEXT I
1570 FOR I=28 TO 5 STEP -4
1580 CALL HCHAR(22,I,152)
1590 CALL SOUND(20,1600,2)
1600 NEXT I
1610 CALL SOUND(100,294,1,440,1,370,1)
1620 CALL SOUND(1000,294,1,440,1,370,1)
1630 FOR I=1 TO 500
1640 NEXT I
1650 GOSUB 1780
1660 PRINT TAB(10);"WELL DONE":::
1670 PRINT TAB(6);"ANOTHER GAME? (Y/N)":::
1680 CALL KEY(0,K,S)
1690 IF S=0 THEN 1680
1700 IF K=89 THEN 350 ELSE 1310
1710 CALL HCHAR(R,C,100)
1720 FOR I=1 TO 500
1730 NEXT I
1740 GOSUB 1780
1750 REM LOSE AGAIN
1760 PRINT " YOU RAN OUT OF OXYGEN":::
1770 GOTO 1670
1780 CALL COLOR(1,4,4)
1790 CALL SCREEN(4)
1800 CALL CLEAR
1810 RETURN

```

TELEPHONE: CAMBERLEY
683686

dps LTD.

337, LONDON ROAD, CAMBERLEY, SURREY

10 BLANK DATA CASSETTES

COMPUTER GRADE TAPE
SUITABLE FOR HOME COMPUTERS
2 WHITE LABELS PLUS 1 INLAY CARD
SUPPLIED FREE

CUT ALONG THIS LINE

1 PACK OF 10 DATA TAPES WITH LIBRARY BOXES
PRICE: £5.75 INC. VAT
POST & PACKAGING FREE.

NAME.....
ADDRESS.....
I ENCLOSE CHEQUE P/O FOR.....

IN-CASSETTE COPYING SERVICE ALSO AVAILABLE

GALAXY

Number One For Texas

EXTENDED BASIC	69.95	SPEECH EDITOR	18.50
PARSEL	26.50	TERMINAL EMULATOR 2	99.95
CHESS	35.50	EDITOR ASSEMBLER	69.95
ADVENTURE/PIRATE	26.50	DRAGON MIX	24.95
THE COUNT	18.50	ALIEN ADDITION	24.95
35 PROGRAM BOOK	5.95	KIDS & TI99/4A	14.36

ALL ORDERS OVER £25 RECEIVE A £1 GIFT VOUCHER

99er MAGAZINE AVAILABLE INDIVIDUALLY
OR BY SUBSCRIPTION

LANTERN

CASSETTE SOFTWARE

DADDIES HOT ROD. (Basic)	Drive Dads Hot Rod into the countryside through sheep fields, woods & over rivers missing the obstacles in the way. £5.95
TROLL KING. (Basic)	An adventure using a full 16K program that accepts English sentences. £6.50
BLASTEROIDS (Ext Basic)	One of the fastest versions available of the classic arcade game. £9.95

PLUS MANY MORE PROGRAMMES

Remember! There are no extra charges at Galaxy.
All prices include VAT and postage. Send SE for full list. Send cheque with order or phone or use Barclaycard or Access.

VISA

60 High Street, Maidstone Kent
Telephone: (0622) 682575 & 679265

Access

Down in the garden there's a battle among the lettuces

How it works

100-290 set colours, define characters
300-340 draw fence
350-440 set score to 0; produces random positions for lettuces, checking position is not already occupied. Number of lettuces can be changed in line 380
450-580 decide random start position for snail and slug, making sure these are not already occupied
590-620 main loop directing program to 'move snail' and 'move slug' subroutines
780-810 get character from screen to determine whether planned move will cross trail, hit fence or eat lettuce
820-860 draw new position of snail then return to main loop
870-1020 calculate scores depending on move made by snail
1030-1170 move slug routine
1180-1210 get character from screen to determine whether planned move will cross a trail, hit fence or eat a lettuce
1220-1260 draw new position of slug; return to main loop
1270-1420 calculate scores depending on move made by slug
1430-1600 subroutine to check whether slug or snail has scored 30 points or more. If winner is decided game ends with message and a chance to try again
1610-1670 subroutine to print statements and scores

In this game for two players you are the slug and your opponent is the snail. The setting is a vegetable patch where you must attempt to eat as many lettuces as you can to gain points.

The problem is that you both leave a sticky trail which neither of you can cross — if you do your opponent will score a point.

At the start of the game the vegetable patch appears, surrounded by its fence. Thirty lettuces sprout up at random. The position of the slug and snail are also randomly assigned.

As these creatures move about a blue trail appears for the slug and a red trail for the snail. When a lettuce is eaten — by guiding your slug/snail onto it — you score two points and your score appears on the allotment fence.

If you cross your own or your opponent's trail, bump into the fence or your opponent, then one point is added to your opponent's score. The game ends with victory for the first player to win 30 points.

Your vegetable patch may seem peaceful, but your lettuces are under attack by a slug and a snail. Choose your side in this two-player game for the TI-99/4A by Jackie Willis

```
100 REM SLUGS AND SNAILS
110 REM BY J.V. WILLIS
120 REM define characters
130 CALL CLEAR
140 FOR I=1 TO 11
150 CALL COLOR(I,16,2)
160 NEXT I
170 CALL COLOR(12,2,2)
180 CALL COLOR(13,3,2)
190 CALL COLOR(14,9,2)
200 CALL COLOR(15,6,2)
210 CALL COLOR(16,11,2)
220 CALL SCREEN(2)
230 CALL CHAR(128,"185A7E7E7E7E3C18")
240 CALL CHAR(152,"9966669999666699")
250 CALL CHAR(136,"00000C0C5E9EFEFF")
260 CALL CHAR(137,"5500DB2424DB0055")
270 CALL CHAR(144,"4224183C3C3C3C18")
280 CALL CHAR(145,"A500DB0000DB00A5")
290 CALL CLEAR
300 REM SET UP SCREEN
310 CALL HCHAR(1,1,152,32)
320 CALL VCHAR(2,1,152,23)
330 CALL VCHAR(1,32,152,23)
340 CALL HCHAR(24,1,152,32)
350 SCORE1=0
360 SCORE2=0
370 RANDOMIZE
380 FOR L=1 TO 30
390 N1=INT(RND*29)+2
400 N2=INT(RND*21)+2
410 CALL GCHAR(N2,N1,GET)
420 IF GET=128 THEN 390
430 CALL HCHAR(N2,N1,128)
440 NEXT L
450 X=INT(RND*29)+2
460 Y=INT(RND*21)+2
470 CALL GCHAR(Y,X,GET)
480 IF GET=128 THEN 450
490 R=X
500 S=Y
510 CALL HCHAR(Y,X,136)
520 X1=INT(RND*29)+2
530 Y1=INT(RND*21)+2
540 CALL GCHAR(Y1,X1,GET)
550 IF (GET=128)+(GET=136) THEN 520
560 R1=X1
570 S1=Y1
580 CALL HCHAR(Y1,X1,144)
```

Hints on conversion

CALL CLEAR clears screen
CALL SCREEN(B) sets screen colour according to value of B
CALL COLOR(A,B,C) determines colours of characters, where A is the character set, B becomes the colours of those characters and C the background colour
CALL CHAR(D,"E") command used for re-defining characters, where D is the ASCII character code and E the hexadecimal code
CALL HCHAR/VCHAR/Y,X,Z,N will produce character Z on the screen starting at row Y in column X and repeating N times horizontally if CALLING HCHAR, or vertically if CALLING VCHAR. The Texas graphics screen is 24 rows by 32 columns, starting at 1,1 in the top left corner
CALL KEY(O,K,S) scans the keyboard to find whether a key K has been pressed
CALL GCHAR(Y,X,Z) gets character Z from screen in row Y and column X
CALL SOUND(DUR,FR,V) produces a sound of duration DUR milliseconds, at a frequency of FR (or a noise if -FR) of volume V

Movements of the slug and snail are controlled from the keyboard — with the alpha lock on — as follows:

Slug:
S left
D right
E up
X down

Snail:
K left
L right
O up
comma down

If your opponent is holding down a key when it's your turn, the game comes to a temporary halt until the key is released. This is so that no player takes more turns than he or she is entitled to!

Variables

SCORE1 snail's score
SCORE2 slug's score
N1,N2 random co-ordinates of lettuces
X,Y co-ordinates of snail
X1,Y1 co-ordinates of slug
R,S co-ordinates of snail's trail
R1,S1 co-ordinates of slug's trail
ROW,COL co-ordinates for printing scores and messages


```

590 REM MAIN LOOP
600 GOSUB 640
610 GOSUB 1040
620 GOTO 600
630 REM MOVE SNAIL
640 CALL KEY(0,K,STATUS)
650 IF K=79 THEN 700
660 IF K=44 THEN 720
670 IF K=76 THEN 740
680 IF K=75 THEN 760
690 GOTO 640
700 Y=Y-1
710 GOTO 770
720 Y=Y+1
730 GOTO 770
740 X=X+1
750 GOTO 770
760 X=X-1
770 IF (Y<1)+(Y>24)+(X<1)+(X>32) THEN 860
780 CALL GCHAR(Y,X,GET)
790 IF GET=32 THEN 820
800 IF GET=128 THEN 870
810 GOTO 950
820 CALL HCHAR(Y,X,136)
830 CALL HCHAR(S,R,137)
840 S=Y
850 R=X
860 RETURN
870 SCORE1=SCORE1+2
880 CALL SOUND(50,-6,2)
890 N$="SNAIL:"&STR$(SCORE1)
900 ROW=24
910 COL=1
920 GOSUB 1630
930 GOSUB 1440
940 GOTO 820
950 SCORE2=SCORE2+1
960 N$="SLUG:"&STR$(SCORE2)
970 CALL SOUND(100,-2,2)
980 ROW=1
990 COL=20
1000 GOSUB 1630
1010 GOSUB 1440
1020 GOTO 820
1030 REM MOVE SLUG
1040 CALL KEY(0,K,STATUS)
1050 IF K=69 THEN 1100
1060 IF K=88 THEN 1120
1070 IF K=68 THEN 1140
1080 IF K=83 THEN 1160
1090 GOTO 1040
1100 Y1=Y1-1
1110 GOTO 1170
1120 Y1=Y1+1
1130 GOTO 1170
1140 X1=X1+1
1150 GOTO 1170
1160 X1=X1-1
1170 IF (Y1<1)+(Y1>24)+(X1<1)+(X1>32)
    THEN 1260
1180 CALL GCHAR(Y1,X1,GET)
1190 IF GET=32 THEN 1220
1200 IF GET=128 THEN 1270
1210 GOTO 1350
1220 CALL HCHAR(Y1,X1,144)
1230 CALL HCHAR(S1,R1,145)
1240 S1=Y1
1250 R1=X1
1260 RETURN
1270 SCORE2=SCORE2+2
1280 CALL SOUND(50,-6,2)
1290 N$="SLUG:"&STR$(SCORE2)
1300 ROW=1
1310 COL=20
1320 GOSUB 1630

```

```

1330 GOSUB 1440
1340 GOTO 1220
1350 SCORE1=SCORE1+1
1360 CALL SOUND(100,-2,2)
1370 N$="SNAIL:"&STR$(SCORE1)
1380 ROW=24
1390 COL=1
1400 GOSUB 1630
1410 GOSUB 1440
1420 GOTO 1220
1430 REM CHECK FOR WINNER
1440 IF SCORE1>29 THEN 1470
1450 IF SCORE2>29 THEN 1510
1460 GOTO 1610
1470 CALL HCHAR(Y,X,136)
1480 CALL HCHAR(S,R,137)
1490 N$=" WELL DONE! THE SNAIL HAS WON."
1500 GOTO 1540
1510 CALL HCHAR(Y1,X1,144)
1520 CALL HCHAR(S1,R1,145)
1530 N$=" WELL DONE! THE SLUG HAS WON."
1540 ROW=12
1550 COL=2
1560 GOSUB 1630
1570 PRINT :
1580 INPUT "PLAY AGAIN?(Y/N) " : A$
1590 IF A$="Y" THEN 290
1600 END
1610 RETURN
1620 REM PRINTING SCORES AND MESSAGES
1630 FOR I=1 TO LEN(N$)
1640 C=ASC(SEG$(N$,I,1))
1650 CALL HCHAR(ROW,(COL-1)+I,C)
1660 NEXT I
1670 RETURN

```

EDITORIAL ASSISTANT

Computing today

Due to promotion, *Computing Today* is seeking an Editorial Assistant. The person we are looking for is an accurate worker with a sense of humour who will be able to maintain the high standards of the magazine while keeping to deadlines. Knowledge of program assessment, development and debugging is required, both in BASIC and machine code/assembly language (preferably for the Z80 and/or 6502). Applicants should be able to express themselves in English but no journalistic experience is necessary: training in technical journalism will be provided.

Familiarity with other languages, particularly Pascal and FORTH, would be an advantage, as would a general knowledge of electronics.

Salary is negotiable depending on age and experience. All applications will be treated in strictest confidence. To apply, send a CV to:

CT Vacancy,
Ron Harris, Managing Editor,
Argus Specialist Publications Ltd,
145 Charing Cross Road,
London WC2H 0EE.

Warning: cigarettes can damage your home, your guests ... and your robot

How it works

110-230 main routine
250 initialisation
260-420 new position for robot
430-440 check for obstacle, cigarette
450 reduce cigarette total by one
460-490 increase score by 10, reduce by rooms destroyed multiplied by 25, prevent negative score
500-540 set flag for score routine, call cigarette plotting and robot position routines, restore last robot co-ordinates
550 crash
560-690 robot sleeps while three extra cigarette ends are plotted, end robot move routine, plot new robot position, print new score if flag set, unset flag
700-760 produce random position for cigarette end, preventing it landing on occupied square; give square a "cigarette value", plot cigarette end, increase cigarette total by one
770-800 store robot co-ordinates, take over co-ordinate variables
810 calls cigarette plot/unplot routine
820-830 restore robot variables
840-920 if 10 cigarettes in room one place chair in each doorway and give doorways "chair values", remember (890) which room, call subroutine for dimensions, call burning subroutine, set cigarette total to zero
930-1370 subroutines for the other five rooms
1390-1490 burning routine scans room, left to right, painting and erasing flame, leaving ash behind. 1440: burning noise
1500 tests to see if robot destroyed
1510-1760 game end messages, reset, return to start if any key pressed
1770 returns from burning room subroutine if robot still alive
1780-1950 subroutine to decide which room has had change in cigarette total, read dimensions of room to be burned
1960 increases factor which reduces score
1980-2070 define characters
2080-2160 set up colours
2160-2220 title page, instructions
2230-2240 player inputs number of cigarettes, figure stored
2250-2750 draw walls, give every square an "obstacle value", print doors, give "clear space" values to squares either side of doorways
2760-2930 plot 30 chairs, 30 guests in unoccupied squares.
2870 ensures men and women are plotted alternately
2940-2970 first position of robot
2980-3030 print score indicator, start

Careless smokers cause fires. Can your heroic robot collect the cigarette ends before your party turns to ashes? Find out with Tony Garner's program for the standard TI-99/4A

It's a great party, but your guests can't seem to find the ashtrays.

Luckily, you have a robot which should collect the cigarette ends — guided by you and your TI-99/4A — before they become a fire hazard.

If a room contains 10 cigarette ends the heat causes it to burst into flames and, if the robot is trapped in the room, he perishes and the game comes to an end.

Otherwise he carries on with his heroic mission in the other rooms until he has cleared up the number of cigarette ends you selected when the game began.

Each time he bumps into a guest, a wall of furniture his circuits are temporarily disrupted and he becomes inactive while three extra cigarettes are dropped.

If four rooms are destroyed the game becomes a question of survival — the player's original figure is ignored while the demoralised robot tries weakly to keep two rooms, and finally one room, clear.

There is no limit to the number of cigarettes that can be requested but, for a competitive game, it would be best to increase them gradually from about 30.

Regardless of average, the best score is the highest. But the average can be increased by intentional crashes. These increase your possible score without raising the number by which it is divided.

The range of the robot can be altered by changing the total of GOSUB 260 lines between 110 and 220.

The game can be made more dangerous by adding more GOSUB 710 lines in the same place or between lines 560 and 590.

Variables

BO number of cigarettes to be dropped
M number of rooms burned down
V,W store robot's vertical (R), horizontal (C) co-ordinates
D ASCII value of key pressed
E cigarette total
TOT basic score
SC score after adjustment
FL flag to show score has changed
CRS current score as string
P,Q cigarette's vertical, horizontal co-ordinates
RR,CC stores for R,C
S(X) total of cigarettes in room (X)
SP room
L(1),L(2)/K(1),K(2) horizontal/vertical dimensions of room to be burned
AV score divided by BO
HSC high score
BAV average for high score
BOA store for BO
ST indicates nature of keypress: same, different, none
A character to be defined
NS hexadecimal string defining character
GR graphics character for male or female guest

```
100 GOTO 240
110 GOSUB 260
120 BO=BO-1
130 IF BO=0 THEN 1510
140 GOSUB 260
150 GOSUB 260
160 GOSUB 260
170 GOSUB 710
180 GOSUB 260
190 IF M=5 THEN 140
200 GOSUB 260
210 IF M=4 THEN 140
220 GOSUB 260
230 GOTO 110
240 DIM MEM(22,28)
250 GOTO 1980
260 V=R
270 W=C
280 CALL KEY(0,D,ST)
290 CALL SOUND(-75,262,2)
300 IF D=88 THEN 350
310 IF D=68 THEN 370
320 IF D=69 THEN 390
330 IF D=83 THEN 410
340 GOTO 700
350 R=R+1
360 GOTO 430
370 C=C+1
```

```
380 GOTO 430
390 R=R-1
400 GOTO 430
410 C=C-1
420 GOTO 430
430 IF MEM(R,C)=3 THEN 530 ELSE 440
440 IF MEM(R,C)=2 THEN 450 ELSE 600
450 E=-1
460 TOT=TOT+10
470 SC=TOT-(25*M)
480 IF SC<10 THEN 490 ELSE 500
490 SC=0
500 FL=1
510 GOSUB 1780
520 GOTO 600
530 R=W
540 C=W
550 CALL SOUND(-250,165,2,1047,2,1760,2)
560 GOSUB 710
570 GOSUB 710
580 GOSUB 710
```

Hints on conversion

CALL KEY returns ASCII value in the variable D

CALL SOUND generates note or noise in following order: duration (negative values enable program to proceed during sound), note value (110 is low, 262 is middle C), volume (0 is loud, 30 is quietest). A negative value from 1 to 8 produces noise: three notes and one noise can be bracketed for same duration

CALL HCHAR plots character by row (1-24), column (1-32), ASCII value and, if needed, number of repeats after and including first position, horizontally

CALL VCHAR as above, but vertical

CALL COLOR sets character set (multiples of eight from 32 to 159), foreground colour (1-16) and background colour (1-16): 1 transparent, 2 black, 5 dark blue, 7 dark red, 10 light red, 12 light yellow, 13 dark green, 14 magenta

The data in line 1950 will need to be adjusted to the new dimensions of the rooms, which will be set, depending on screen size, in lines 2260-2620. Doors will also have to be altered.

Graphics designed in 2040-2070 are: 42 robot, 120 flame, 128 ash, 136 cigarette end, 144 chair, 104 female guest, 96 male guest, 152 top wall, 154 bottom wall, 155 interval wall horizontal, 156 central corridor, 157 vertical wall.

The program sits well inside genuine 16K, but the array DIMensioned in 240 consumes a lot of memory.

The TI does not allow multiple statement lines, but these may well be required for other machines with less available memory.

The GOSUB structure should make life easy for the BBC micro users to make a straightforward adaptation.

TI-99/4A PROGRAM

```

590 RETURN
600 MEMR,C)=1
610 MEMV,W)=0
620 CALL HCHAR(V,W,32)
630 CALL HCHAR(R,C,42)
640 IF FL=1 THEN 650 ELSE 700
650 CR=STR$(SC)
660 FOR J=1 TO LEN(CR)
670 CALL HCHAR(24,13+J,ASC(SEG$(CR),J,1)))
680 NEXT J
690 FL=0
700 RETURN
710 P=INT(RND*16)+5
720 Q=INT(RND*22)+4
730 IF (MEMP,Q)=1)+(MEMP,Q)=2)+(MEMP,Q)=3)+(MEMP,Q)=4) THEN 710
740 MEMP,Q)=2
750 CALL HCHAR(P,Q,135)
760 E=1
770 RR=R
780 CC=C
790 R=P
800 C=Q
810 GOSUB 1700
820 R=RR
830 C=CC
840 IF S(1)=10 THEN 850 ELSE 930
850 CALL HCHAR(15,5,144)
860 MEM(5,5)=3
870 CALL HCHAR(12,9,144)
880 MEM(12,9)=3
890 SP=1
900 GOSUB 1910
910 GOSUB 1390
920 S(1)=0
930 IF S(2)=10 THEN 940 ELSE 1020
940 CALL HCHAR(6,15,144)
950 MEM(6,15)=3
960 CALL HCHAR(12,9,144)
970 MEM(12,9)=3
980 SP=2
990 GOSUB 1910
1000 GOSUB 1390
1010 S(2)=0
1020 IF S(3)=10 THEN 1030 ELSE 1110
1030 CALL HCHAR(15,5,144)
1040 MEM(15,5)=3
1050 CALL HCHAR(20,15,144)
1060 MEM(20,15)=3
1070 SP=3
1080 GOSUB 1910
1090 GOSUB 1390
1100 S(3)=0
1110 IF S(4)=10 THEN 1120 ELSE 1200
1120 CALL HCHAR(6,15,144)
1130 MEM(6,15)=3
1140 CALL HCHAR(9,26,144)
1150 MEM(9,26)=3
1160 SP=4
1170 GOSUB 1910
1180 GOSUB 1390
1190 S(4)=0
1200 IF S(5)=10 THEN 1210 ELSE 1290
1210 CALL HCHAR(20,15,144)
1220 MEM(20,15)=3
1230 CALL HCHAR(18,21,144)
1240 MEM(18,21)=3
1250 SP=5
1260 GOSUB 1910
1270 GOSUB 1390
1280 S(5)=0
1290 IF S(6)=10 THEN 1300 ELSE 1390
1300 CALL HCHAR(9,26,144)
1310 MEM(9,26)=3
1320 CALL HCHAR(18,21,144)
1330 MEM(18,21)=3
1340 SP=6
1350 GOSUB 1910
1360 GOSUB 1390
1370 S(6)=0
1380 RETURN
1390 FOR Y=L(1) TO L(2)
1400 FOR Z=K(1) TO K(2)
1410 CALL HCHAR(Y,Z,120)
1420 CALL HCHAR(Y,Z,32)
1430 MEM(Y,Z)=3
1440 CALL SOUND(-750,-5,15)
1450 CALL HCHAR(Y,Z,120)
1460 CALL HCHAR(Y,Z,32)
1470 CALL HCHAR(Y,Z,120)
1480 NEXT Z
1490 NEXT Y
1500 IF MEMR,C)=3 THEN 1510 ELSE 1770
1510 CALL CLEAR
1520 AV=SC/BOR
1530 IF SC>HSC THEN 1540 ELSE 1560
1540 HSC=SC
1550 BAV=AV
1560 PRINT TAB(10);"THAT'S IT";"...";"FINAL SCORE=";SC;"WITH";BOR;"CIGARETTES";"ORI
GINALLY TO BE";"CLEARED AWAY";
1570 PRINT "YOUR AVERAGE";AV;
1580 PRINT "REMEMBER YOU CAN";"IMPROVE YOUR AVERAGE";"IF YOU DARE BY CAUSING";
"OCCASIONAL ACCIDENTS";
1590 PRINT "BEST SCORE SO FAR";HSC;"AT AN AVERAGE OF";BAV;
1600 PRINT "PRESS ANY KEY TO PLAY AGAIN";
1610 CALL KEY$(0,D,ST)
1620 IF ST=0 THEN 1610
1630 CALL CLEAR
1640 PRINT "GETTING UP NOW";
1650 FOR R=1 TO 22
1660 FOR C=1 TO 20
1670 MEMR,C)=0
1680 NEXT C
1690 NEXT R
1700 SC=0
1710 TOT=0
1720 M=0
1730 FOR J=1 TO 6
1740 S(J)=0
1750 NEXT J
1760 GOTO 240
1770 RETURN
1780 IF (R(15)<C(9)) THEN 1790 ELSE 1800
1790 S(1)=S(1)+E
1800 IF (R(15)<C(9))<C(15)) THEN 1810 ELSE 1820
1810 S(2)=S(2)+E
1820 IF (R(15)<C(15))<C(15)) THEN 1830 ELSE 1840
1830 S(3)=S(3)+E
1840 IF (R(9)<C(15))<C(15)) THEN 1850 ELSE 1860
1850 S(4)=S(4)+E
1860 IF (R(9)<C(15))<C(21)) THEN 1870 ELSE 1880
1870 S(5)=S(5)+E
1880 IF (R(9)<C(21))<C(21)) THEN 1890 ELSE 1900
1890 S(6)=S(6)+E
1900 RETURN
1910 RESTORE 1950
1920 FOR J=1 TO SP
1930 READ L(1),L(2),K(1),K(2)

```

```

1940 NEXT J
1950 DATA 3,14,4,8,3,14,10,14,16,21,4,14,3,8,16,27,10,21,16,20,10,21,22,27
1960 M=M+1
1970 RETURN
1980 RANDOMIZE
1990 RESTORE 2040
2000 FOR J=1 TO 12
2010 READ A,N#
2020 CALL CHAR(A,N#)
2030 NEXT J
2040 DATA 42,92FE92387CFE44,120,0010101A95D7F7F,120,2041021009401200
2050 DATA 136,00000075,144,003C242424E4242,104,1010121E103C1010,96,101040701000
0010
2060 DATA 152,00000000000000FF,154,FF,155,FF0000A0000000FF
2070 DATA 156,0101010101010101,157,9101010101010101
2080 CALL COLOR(2,5,1)
2090 CALL COLOR(9,2,1)
2100 CALL COLOR(10,14,1)
2110 CALL COLOR(12,10,12)
2120 CALL COLOR(14,7,1)
2130 CALL COLOR(15,13,1)
2140 CALL SCREEN(10)
2150 CALL CLEAR
2160 PRINT TAB(3);" * ROBOT FIREWATCH * "
2170 PRINT " * COLLECTS DROPPED DOG-ENDS * "
2180 PRINT " * USING THE CURSOR KEYS YOU * " "HAVE TO STEER * " "AROUND SIX ROOMS IN * "
" A HOUSE WHERE A PARTY "
2190 PRINT " IS BEING HELD * " "IF * HITS A GUEST * " "OR A WALL OR A CHAIR * " "HIS CI
RCUITS TAKE TIME * " "TO RECOVER * "
2200 PRINT " IF TEN CIGARETTE ENDS * " "ARE CONCENTRATED IN ONE ROOM * " "NOTHING CAN P
REVENT A FIRE * " "POINTS LOST * "
2240 DATA 6,14,6,15,6,16,12,9,12,9,12,10,10,20,10,21,10,22
2250 DATA 8,26,9,26,10,26,14,5,15,5,16,5,20,14,20,15,20,16
2260 FOR I=1 TO 30
2270 R=INT(RND*16)+5
2280 C=INT(RND*22)+4
2290 IF (MEMR,C)=3)+(MEMR,C)=4) THEN 2770
2300 MEMR,C)=3
2310 CALL HCHAR(R,C,144)
2320 NEXT I
2330 FOR I=1 TO 30
2340 R=INT(RND*16)+5
2350 C=INT(RND*22)+4
2360 IF (MEMR,C)=3)+(MEMR,C)=4) THEN 2840
2370 IF INT(1/2)=1/2 THEN 2880 ELSE 2900
2380 GR=96
2390 GOTO 2910
2400 GR=104
2410 MEMR,C)=3
2420 CALL HCHAR(R,C,GR)
2430 NEXT I
2440 R=5
2450 C=26
2460 MEMR,C)=1
2470 CALL HCHAR(R,C,42)
2480 N#="SCORE="
2490 FOR J=1 TO LEN(N#)
2500 CI=ASC(SEG$(N#,J,1))
2510 CALL HCHAR(24,7+J,CI)
2520 NEXT J
2530 GOTO 110

```

```

2210 PRINT "AND WHEN ONLY TWO ROOMS * " "REMAIN * " "S
MORALE FALLS * " "SKILL LEVEL DE CIDES * "
2220 PRINT "HOW MANY CIGARETTES * " "ARE DROPPED BEFORE
THE GAME ENDS * " "INPUT YOUR NUMBER NOW * "
2230 INPUT B0
2240 BOR=B0
2250 CALL CLEAR
2260 CALL SCREEN(15)
2270 FOR C=4 TO 27
2280 CALL HCHAR(C,152)
2290 MEM(C,3)=3
2300 NEXT C
2310 FOR C=4 TO 27
2320 CALL HCHAR(22,C,154)
2330 MEM(22,C)=3
2340 NEXT C
2350 FOR R=3 TO 21
2360 CALL HCHAR(R,3,157)
2370 MEM(R,3)=3
2380 NEXT R
2390 FOR R=3 TO 21
2400 CALL HCHAR(R,28,157)
2410 MEM(R,28)=3
2420 NEXT R
2430 FOR C=4 TO 14
2440 CALL HCHAR(15,C,155)
2450 MEM(15,C)=3
2460 NEXT C
2470 FOR C=16 TO 27
2480 CALL HCHAR(9,C,155)
2490 MEM(9,C)=3
2500 NEXT C
2510 FOR R=3 TO 21
2520 CALL HCHAR(R,15,156)
2530 MEM(R,15)=3
2540 NEXT R
2550 FOR R=3 TO 14
2560 CALL HCHAR(R,9,157)
2570 MEM(R,9)=3
2580 NEXT R
2590 FOR R=10 TO 21
2600 CALL HCHAR(R,21,157)
2610 MEM(R,21)=3
2620 NEXT R
2630 CALL HCHAR(6,15,32)
2640 CALL HCHAR(12,9,32)
2650 CALL HCHAR(10,21,32)
2660 CALL HCHAR(9,26,32)
2670 CALL HCHAR(15,5,32)
2680 CALL HCHAR(20,15,32)
2690 RESTORE 2740
2700 FOR I=1 TO 10
2710 READ R,C
2720 MEMR,C)=4
2730 NEXT I

```

AGF PROGRAMMABLE JOYSTICK INTERFACE for Spectrum or ZX81

ABOUT OUR PROGRAMMABLE INTERFACE

Surpassing the outstanding specification of our Interface Module II which still offers the best software support at its price, a joystick interface that is compatible with ALL SOFTWARE through its unique hardware programmable design.

The interface plugs on to the rear connector of your ZX81 or ZX Spectrum.

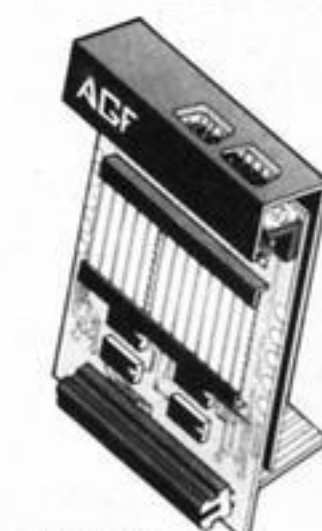
Quick clip-on connections, which are configured from a reference card supplied, allow you to define which of the forty keys are simulated by which action of the joystick. A pack of ten Quick Reference Programming Cards makes setting for your favourite games even easier. These can be filled in to show at a glance the configuration required and stored in the cassette case of the particular game. When you change to a game using different keys the module is re-programmed in a few seconds.

As with our Interface Module II the Programmable Interface accepts all standard joystick sticks that are Atari-compatible. Two sockets are available which are connected together for two-player games which use the same keys for both players.

The interface resides in the keyboard address space and does not affect its operation or interfere with any other add-ons. A rear extension edge connector will accommodate expansion of your system.

The unique AGF key simulation principle makes it extremely easy to incorporate joystick action in your own programs. All eight directions and fire are read by simple BASIC.

With every order comes a free demonstration program called 'Video Graffiti' plus a full set of instructions.



KEY FEATURES

- Programmable design gives TOTAL software support.
- Accepts Atari, Competition Pro, Wico, Starfighter, Quick Shot, Le Stick etc.
- Rear extension connector for all other add-ons.
- Free demo program and instructions.

PACKAGE CONTENTS SUPPLIED

- Programmable Interface Module as illustrated, complete with clip-on programming leads.
- Self adhesive programming chart detailing how to define which key is simulated by UP, DOWN, LEFT, RIGHT, and FIRE. This can be fixed on to the case of your computer or if preferred the protective backing can be left on. The chart is made of a very durable reverse printed plastic and is extremely easy to read.
- One pack of ten Quick Reference Programming Cards for at-a-glance setting to your games requirements. The card allows you to mark the configuration in an easy to read fashion with space to record the software title and company name.
- Video Graffiti demonstration program which is written totally in BASIC to illustrate how all eight directions and fire can be read. This is also a useful high resolution drawing program.
- 12 months guarantee and full written instructions.

JOYSTICKS ATARI CONTROLLERS

FOR USE WITH OUR INTERFACE Module or VIC 20, Commodore 64, Atari VCS, Atari 400, Atari 800

If you require extra joysticks for our original interface module mark order 'OLD' joysticks

ONLY £7.54 inc VAT + P&P

FROM: MR/MRS/MISS			
ADDRESS			
SEND C.W.O. (NO STAMP NEEDED) TO: A.G.F. HARDWARE, DEPT. HCW			
FREEPOST, BOGNOR REGIS, WEST SUSSEX, PO22 9BR			
QTY	ITEM	ITEM PRICE	TOTAL
	PROGRAMMABLE INTERFACE	33.95	
	JOYSTICK(S)	7.54	
	PACK(S) QUICK REFERENCE CARDS	1.00	
ONE	VIDEO GRAFFITI	FREE	
ZX81 ☐ ZX SPECTRUM ☐ Please tick		FINAL TOTAL	
DEALER ENQUIRIES WELCOME		EXPORT PRICES ON APPLICATION	

Show off your skill at the wheel as you race round the forest

Now is your chance to be Roger Clark, Paddy Hopkirk or any of the great rally drivers in my Forest Rallye game.

You must drive your car round the forest road to the finishing line using keys E,S,D and X to steer. Hit trees and you will be penalised and driving out of the forest disqualifies you. Oh, by the way, you have no brakes...

Your score and best score are displayed at the end of the game to prove what a great driver you are.

If you find the control keys awkward to use, change the ASCII values of K in lines 750, 800, 850 and 900 to those of keys you find more comfortable.

How it works

140-300 colour screen, define characters and colours
320-410 print instructions, wait for key press
430-610 set up screen graphics
630-680 set initial variables
700-930 scan keys, calculate new position of car
950-1000 check car position, move car
1020-1080 car off track
1100-1230 finish, show scores and option for another game
1240-1290 car out of forest
1300-1310 data for course
1340-1380 PRINT AT routine

Prove your driving ability with this testing route through a forest. Graham Baldwin gives you the opportunity with his game for the unexpanded TI-99/4A

```
100 REM      *FOREST RALLY*
110 REM
120 REM      *BY GRAHAM BALDWIN*
130 REM
140 CALL CLEAR
150 CALL SCREEN(2)
160 REM DEF GRAPHICS
170 CALL CHAR(96,"0000784CFFFF63")
180 CALL CHAR(97,"00001E32FFFFC6")
190 CALL CHAR(98,"307C74343C387070")
200 CALL CHAR(99,"7070383C34747C30")
210 CALL CHAR(104,"101038387C7C1010")
220 CALL CHAR(112,"FFFFFFFFFFFFFFFF")
230 CALL CHAR(120,"CCCC3333CCCC3333")
240 REM DEF COLOURS
250 FOR I=1 TO 8
260 CALL COLOR(I,16,2)
270 NEXT I
280 CALL COLOR(12,9,16)
290 CALL COLOR(9,14,1)
300 CALL COLOR(10,3,1)
310 BST=0
```

Hints on conversion

Texas (TI) BASIC is fairly standard with two main exceptions. Single statement lines are used and sub-routines for graphics and sound are available, prefixed with CALL, as detailed below.

CALL CLEAR clears screen

CALL SCREEN defines screen colour

CALL CHAR defines a character with a 16-digit hex string from an eight by eight grid. Other machines will use a different method

CALL COLOR (character set, foreground,background) set colour of character

CALL HCHAR (row,column, character code, number of repeats) places character on screen at co-ordinates specified and optionally repeats character horizontally

CALL VCHAR similar to HCHAR but repeats character vertically

CALL GCHAR (row,column,variable) similar to a screen PEEK on other machines. Gives the ASCII code at the given screen co-ordinates

CALL SOUND (D,F,V) gives sound of duration D, frequency F and volume V. Three sounds can be produced at once.

```
320 PRINT TAB(8);"FOREST RALLYE"
330 PRINT TAB(7);"*****"
340 PRINT "USE THE ARROW KEYS (ESDX) TO"
350 PRINT "DRIVE YOUR CAR THROUGH"
360 PRINT "THE FOREST. YOU WILL BE"
370 PRINT "PENALISED FOR LEAVING THE"
380 PRINT "FOREST OR HITTING TREES."
390 PRINT "PRESS ANY KEY TO BEGIN."
400 CALL KEY(0,K,S)
410 IF S=0 THEN 400
420 REM SET UP SCREEN
430 CALL HCHAR(1,1,104,768)
440 FOR I=1 TO 10
450 READ A,B,C
460 CALL HCHAR(A,B,112,C)
470 NEXT I
480 FOR I=1 TO 11
490 READ A,B,C
500 CALL VCHAR(A,B,112,C)
510 NEXT I
520 RESTORE
530 M$="START"
540 RW=2
550 CW=2
560 GOSUB 1340
570 M$="FINISH"
580 RW=8
```

```
590 CW=5
600 GOSUB 1340
610 CALL HCHAR(7,2,120,3)
620 REM INITIAL VARIABLES
630 R=4
640 C=4
650 X=1
660 Y=0
670 CD=96
680 CR=0
690 REM SCAN KEYS
700 CALL KEY(0,K,S)
710 RE=R
720 CE=C
730 IF S=0 THEN 950
740 CALL SOUND(100,-2,3)
750 IF K<>69 THEN 800
760 Y=-1
770 X=0
780 CD=99
790 GOTO 950
800 IF K<>88 THEN 850
810 Y=1
820 X=0
830 CD=98
840 GOTO 950
850 IF K<>83 THEN 900
860 Y=0
```

Variables

R,C row, column of car
RE, RC erase row and column of car
X,Y decide direction of car
CD redefines car for different direction
CR number of crashes
BST best score

TI-99/4A PROGRAM

```

870 X=-1
880 CD=97
890 GOTO 950
900 IF K<>68 THEN 950
910 Y=0
920 X=1
930 CD=96
940 REM MOVE CAR
950 R=R+Y
960 C=C+X
970 IF (R>24)+(R<1)+(C>32)+(C<1) THEN 1240
980 CALL GCHAR(R,C,OFF)
990 CALL HCHAR(RE,CE,112)
1000 CALL HCHAR(R,C,CD)
1010 REM CAR OFF TRACK
1020 IF OFF=112 THEN 700
1030 IF OFF=120 THEN 1100
1040 CR=CR+1
1050 CALL SCREEN(7)
1060 CALL SOUND(-400,-7,2)
1070 CALL SCREEN(2)
1080 GOTO 700
1090 REM FINISH
1100 FOR I=110 TO 1100 STEP 110
1110 CALL SOUND(-400,I,1)
1120 NEXT I
1030 IF OFF=120 THEN 1100
1040 CR=CR+1
1050 CALL SCREEN(7)
1060 CALL SOUND(-400,-7,2)
1070 CALL SCREEN(2)
1080 GOTO 700
1090 REM FINISH
1100 FOR I=110 TO 1100 STEP 110
1110 CALL SOUND(-400,I,1)
1120 NEXT I

```

```

1130 CALL CLEAR
1140 REM SCORE COUNT
1150 IF (BST=0)+(BST>CR) THEN 1160 ELSE 1170
1160 BST=CR
1170 PRINT TAB(6);"YOU HIT ";CR;" TREES":
1180 PRINT TAB(4);"BEST RUN SO FAR IS ";BST:
1190 PRINT TAB(7);"TRY AGAIN? (Y/N)":
1200 CALL KEY(0,K,S)
1210 IF S<1 THEN 1200
1220 IF K=89 THEN 430
1230 IF K=78 THEN 1320 ELSE 1190
1240 FOR I=1100 TO 110 STEP -110
1250 CALL SOUND(-400,I,1)
1260 NEXT I
1270 CALL CLEAR
1280 PRINT "YOU DROVE OUT OF THE FOREST.":
1290 GOTO 1190

1300 DATA 4,4,9,3,14,6,4,21,9,8,27,3,12,20,
        6,9,16,3,16,16,13,22,14,15,20,7,5,15,
        4,2

1310 DATA 3,13,2,3,20,2,4,30,5,8,26,5,9,19,
        4,9,15,8,16,29,7,21,13,2,20,12,2,15,6,
        6,8,3,8

1320 END
1330 REM 'PRINT AT' ROUTINE
1340 FOR I=1 TO LEN(M$)
1350 WD=ASC(SEG$(M$,I,1))
1360 CALL HCHAR(RW,CW+I,WD)
1370 NEXT I
1380 RETURN

```

[illegible]

Save us from an alien invasion

How's your aim? Type in James Burton's game for the TI-99/4A and test your skill with a laser gun

Variables

X,Y co-ordinates of alien
SC player's score
EN player's energy (1 to 40)
S pointer for which stars are shining
C pointer for which set of laser bolts can be seen
 The arrays **A()**, **B()**, **E()**, **H()** and **M()** are used to try to speed the program up by making it unnecessary to check if the alien moves off the screen etc.
 The array **F()** is not strictly necessary but is used to store the characters that are **GCHAR**ed. It is quicker to access this array than to use **GCHAR** more than once.

the left, making the stars and alien move right and pressing the equals key moves the turret to the right.

When the space bar is pressed four laser bolts are fired, hopefully smashing the alien into little pieces.

Hints on conversion

CALL CHAR (ASCII) code number, hexadecimal string) Defines characters used in the program. The hexadecimal string describes the pattern to be given to the character with ASCII code stated.

CALL HCHAR (row number, column number, ASCII code, number of repetitions) Places character on screen at row and column specified, and repeat it horizontally the number of times stated. If this fourth value is omitted then the character is displayed only once. The character used is the one whose ASCII code appears in the command.

CALL VCHAR (row number, column number, ASCII code, number of repetitions) Works in the same way as **CALL HCHAR** only repeats vertically instead of horizontally.

CALL GCHAR (row number, column number, numeric, numeric variable) Equivalent of **PEEK**, assigns ASCII code of character at row and column stated to numeric variable.

CALL CLEAR Clears the screen
CALL SCREEN (Colour code) Specifies screen colour, using colour code.

CALL COLOR (character set number, foreground colour code, background colour code) Specifies foreground and background colours for all characters in set stated.

CALL SOUND (duration, frequency, volume) Produces sound with duration, frequency, and volume specified. Duration is in milliseconds, frequency in Hertz (or noise code), and volume in scale from 0 loudest to 30 quietest.

CALL KEY (key unit, k,s) Equivalent of **INKEYS** in other computers, returns code if key pressed in variable k.

In Earth Defence you control a high power gun firing laser bolts. In the night sky an alien flits around and it is your job to shoot it down.

When the program is run, your screen becomes the view from the gun turret.

Pressing 1 moves the turret to

How it works

130-320 scan keyboard, move alien

330-450 move laser bolts, check whether alien is hit

460-640 give attractive display when alien is hit

650-690 decrease energy, check if there is no energy left

700-880 give attractive display when player has zero energy and then shows score

890-1760 this long section sets up the numerous arrays and user defined characters and explains the game's theme

```
100 DIM A(27),B(23),E(7,1),F(27,23),H(10),M(16),Z(1)
110 RANDOMIZE
120 GOSUB 900
130 REM MAIN ROUTINE
140 CALL KEY(3,K,T)
150 IF K=32 THEN 340
160 I=(K-61)-(K-49)
170 IF I=0 THEN 210
180 CALL COLOR(5,2,2)
190 S=S+I+M(S+1)
200 CALL COLOR(S,8,2)
210 X1=RX+SGN(E(0,0)*.5+1)
220 Y1=RY+E(0,1)
230 IF F(X1,Y1) THEN 250
240 CALL GCHAR(Y1,X1,F(X1,Y1))
250 CALL HCHAR(Y,X,F(X,Y))
260 CALL HCHAR(Y1,X1,40+0)
270 X=X1
280 Y=Y1
290 D=D+RND-.7
300 D=D+H(D+1)
310 IF RND<.2 THEN 660
320 IF J THEN 390 ELSE 140
330 REM FIRE ROUTINE
340 CALL COLOR(3,8,2)
350 CALL SOUND(-300,220,0,222,0)
360 J=1
370 C=4
380 I=0
390 CALL COLOR(C,0,2)
400 CALL COLOR(C-1,2,2)
410 IF ABS(F(X,Y)-C*8-20,5)<4.5 THEN 470
420 C=C+1
430 IF C<8 THEN 210
440 J=0
450 GOTO 660
460 REM ALIEN HIT
470 SC=SC+1
480 IF EN<33 THEN 500
490 EN=33
500 EN=EN+6
510 G=INT(EN*.25)
520 CALL HCHAR(24,12,156,G)
530 CALL HCHAR(24,12+G,152+EN-4*G)
540 CALL SOUND(-2000,-7,0)
550 IF C=7 THEN 570
560 CALL COLOR(C,2,2)
570 FOR L=1 TO 20
580 CALL COLOR(2,2,16)
590 CALL COLOR(2,16,2)
600 NEXT L
610 CALL HCHAR(Y,X,F(X,Y))
620 GOSUB 1720
630 J=0
640 GOTO 210
650 REM DECREASE ENERGY
660 EN=EN-1
670 G=INT(EN*.25)
680 CALL HCHAR(24,12+G,152+EN-4*G)
690 IF EN THEN 320
700 REM NO ENERGY LEFT
710 FOR L=1 TO 100
720 G=RND*15+1.5
730 CALL COLOR(RND*16+.5,G,G)
740 CALL SOUND(-500,G*100+110,0)
750 NEXT L
760 CALL CLEAR
770 FOR L=2 TO 7
780 CALL COLOR(L,2,16)
790 NEXT L
800 PRINT "SCORE:"
810 FOR L=1 TO SC/10
820 CALL HCHAR(25-L,12,40,10)
830 NEXT L
840 CALL HCHAR(24-INT(SC/10),12,40,SC-10*INT(SC/10))
850 FOR L=1 TO 2000
860 NEXT L
870 GOSUB 1390
880 GOTO 140
890 REM INITIALIZATION
900 CALL CLEAR
910 CALL SCREEN(8)
920 PRINT " ** EARTH DEFENCE ** " "SHOOT DOWN THE ALIENS USING YOUR HIGH POW
ER ACK-ACK GUN"
930 PRINT "1-ROTATE GUN LEFT" "2-ROTATE GUN RIGHT" "SPACEBAR-FIRE" "WATCH YOUR E
NERGY BANK"
940 INPUT "PRESS ENTER TO PLAY,"A$
950 CALL CLEAR
960 FOR L=3 TO 22
970 A(L+4)=L+4
980 B(L)=L
990 NEXT L
1000 A(6)=7
1010 A(27)=26
```

```
1020 B(2)=3
1030 B(23)=22
1040 E(0,1)=-1
1050 FOR L=1 TO 7
1060 E(L,0)=SGN(E(L-1,1)+E(L-1,0))
1070 E(L,1)=SGN(E(L-1,1)-E(L-1,0))
1080 NEXT L
1090 H(0)=0
1100 H(9)=0
1110 H(10)=0
1120 M(16)=0
1130 M(7)=0
1140 FOR L=48 TO 80 STEP 8
1150 CALL CHAR(L,"000001000040201")
1160 CALL CHAR(L+1,"0100040010204000")
1170 NEXT L
1180 Z(0)=""
1190 Z(1)=""
1200 FOR L=88 TO 144 STEP 8
1210 CALL CHAR(L,"0000001010")
1220 CALL CHAR(L+1,"0000103C3C10")
1230 Z(0)=Z(0)+CHR(L)
1240 Z(1)=Z(1)+CHR(L+1)
1250 NEXT L
1260 Z(0)=Z(0)+Z(0)+Z(0)+Z(0)
1270 Z(1)=Z(1)+Z(1)+Z(1)+Z(1)
1280 DATA 1010303020440202,00607E2120202010,0003043BF0300403,10202020217E6000,41
4122141C10000,00040404047E0601
1290 DATA C0201C0F1C20C0,01067E0404040400
1300 FOR L=48 TO 47
1310 READ A$
1320 CALL CHAR(L,A$)
1330 NEXT L
```

```
1340 DATA 00,C0C0C0C0C0C0C0,F0F0F0F0F0F0F0,F0F0F0F0F0F0F0,FFFFFFF0F0F0F0,FFFFFFF0F0F0F0
1350 FOR L=152 TO 156
1360 READ A$
1370 CALL CHAR(L,A$)
1380 NEXT L
1390 RESTORE 1400
1400 DATA 0,16,2,2,2,2,0,0,2,2,2,2,2,2,2
1410 FOR L=1 TO 15
1420 READ G
1430 CALL COLOR(L,G,2)
1440 NEXT L
1450 CALL COLOR(16,7,10)
1460 CALL CLEAR
1470 FOR L=1 TO 12
1480 G=RND
1490 J=INT(RND*4+10)
1500 Z(G)=SEG$(Z(G),J,20)+SEG$(Z(G),1,J-1)
```

```
1510 PRINT "TAB(3);Z(G)"
1520 NEXT L
1530 CALL VCHAR(23,20,31,196)
1540 CALL VCHAR(23,5,31,4)
1550 CALL VCHAR(23,6,31,2)
1560 CALL VCHAR(1,27,31,2)
1570 CALL VCHAR(23,27,31,4)
1580 FOR L=3 TO 7
1590 FOR G=0 TO 1
1600 G=L*2+G
1610 CALL HCHAR(G-3,G+1,L*8+24)
1620 CALL HCHAR(G-3,32-G,L*8+25)
1630 CALL HCHAR(28-G,G+1,L*8+25)
1640 CALL HCHAR(28-G,32-G,L*8+24)
1650 NEXT G
1660 NEXT L
1670 SC=0
1680 EN=40
1690 CALL HCHAR(24,12,156,10)
1700 B=0
1710 J=0
1720 D=INT(RND*8)
1730 F(16,12)=00
1740 X=16
1750 Y=12
1760 RETURN
```


The double life of Stainless Software's supremo

Most TI-99/4A owners seem to agree that theirs is a wonderful machine — but the big problem is, you can't get software for it easily.

Stephen Shaw is a great fan of the TI, whose solution to the software shortage was to set up his own company to get hold of some. Stainless Software, the UK's largest independent supplier of TI programs, was the result.

Stainless's current catalogue lists 71 programs. Of these, 26 are written by UK programmers under the Stainless label. The other 45 are imported from a selection of North American TI software companies. There's something in there for all tastes: arcade games, mind games, utilities, adventures, even a typing tutor.

Stephen and his wife Cathy run the software business from a corner of the dining room in their 1920s semi at 10 Alstone Road, Stockport. From this modest base they deal with orders, do some duplicating, and produce the Stainless catalogue — which also acts as a kind of newsletter, with information on cassette care, and details of the TI User Group.

During the day, they both work as bank clerks — Stephen at Williams and Glyn's, and Cathy at Barclays.

But though they both work in banks, they aren't exactly laughing all the way to one on the proceeds from Stainless. Their

Britain's biggest independent TI software dealer is a dining-room business. Candice Goodwin talked to Stephen Shaw, who's a bank clerk by day and runs Stainless Software in the evening

turnover is "big enough to register for VAT" — more than £18,000 — but in the first quarter of 1983, Stephen says Stainless made a small loss.

"With only 20,000 TI-99/4A owners in the UK, even my most popular game won't sell more than a couple of thousand copies," he explained. "Fortunately, my interest is in the TI — the sales are really just to support the user. I'm not in it for the money, so I sell at the best price I can — I deliberately keep my profit margins low." Thus speaks a true enthusiast.

Stephen's interest in com-

puting was first awakened 10 years ago when Commodore announced the PET. He said: "I've a fascination for science and maths, and though I'm not an academic sort, I like to apply logical principles. I saw computing as a way to do that."

He sent off for details of the PET — and was "decisively ignored" by Commodore. But Texas Instruments had a store in Manchester where Stephen saw a forerunner of the present TI-99/4A. He borrowed the machine at weekends, and wrote to TI to find out more. "Unlike everyone else, TI did respond to letters", he recalls.

He was attracted to the TI machine on several counts. "It's easy to program it — the BASIC's well arranged. It's easy to hook it up to peripherals. For example, other computers have a difficult disc filing command. It's very easy to use on the TI. And the computer itself is neat and attractive. It sits in the dining room very nicely." In October 1981, he finally got his hands on a TI of his own.

But despite being a charter member of the 99er, an American magazine for TI owners, and a founder member of the TI User Group, Stephen found there was "a lot of information but no software" for the TI.

After sending off to the US for programs for his own use, he decided to try and distribute US software in the UK himself. When he signed his first contract in June last year, with Not Polyoptics, there were a mere 100 TIs in this country.

With numbers like that, large software stocks just weren't on. So Stephen arranged to produce copies from a master tape supplied by the software company, and pay them a fixed percentage of the US price per copy — an arrangement which he still uses today.

But he finds that "the price difference is a problem" in selling software from the US, where consumers are used to paying \$26-27 per program. Sometimes Stephen ends up selling programs for less than their selling price in the US.

Though he's sticking to his existing US suppliers, Stephen is now trying to get hold of more material from UK software writers. He currently has two regular programmers and about 10 or 11 others, most of which he got in contact with through the appeal for programs in his catalogue.

Stephen has one regular dealer, Galaxy Software, but he's wary of retailers on the whole because he says: "They want huge discounts. Whereas in the US they'll take 30 per cent of the selling price, over here 50-60 per cent is the norm. In order to keep prices down, I can't afford to give much discount."

"And I think the TI-99/4A itself is oversold by dealers — they don't warn customers that you can't run arcade-type games in TI BASIC. Whereas in my mail-order catalogue, I can warn people that TI BASIC games will be slow."

"My biggest problem is the TI itself. It's very good at what it does, but the unexpanded machine won't do what people often buy it for — playing arcade games."

To help people see the light about this much misunderstood computer, Stephen is currently writing a book for Phoenix Publishing describing the TI. Though he's only written about 60,000 words so far, he plans to have it finished for Christmas, and says it will contain "a lot of information which isn't available elsewhere."

He's also looking forward to the new TI computer, which he "definitely will" be selling software for. A TI fan to the last, he believes this machine "will be superior to the BBC — after all, it will be faster, have more features, and more RAM. If they can bring it out by Christmas, it could wipe the English market clean."



Stephen Shaw — a true TI enthusiast

There's heaps of trouble on the grisly grid

As my game begins you find yourself precariously perched on a grid — and that's the only thing between you and the gaping mouth of an atomic reactor.

Dotted about the grid are a number of piles of atomic dust, and your task is to collect them. Every time you succeed, your score increases.

But there's a catch — in fact, there are two. As you move, random static appears along your route, and if you touch it, instant death follows. Falling off the grid into the reactor won't do you much good either.

The game includes instructions, an initialised high score routine, and a selectable skill level which determines the number of dustpiles available.

If you collect more than 20 piles, a new game starts and your score increases. Off you go — and watch your step.

If you think you can keep your balance, try your skill in Frank O'Connor's electrifying game for the TI-99/4A



```

10 CALL CLEAR
20 REM *****
30 REM *!!#ELECTRON!! *
40 REM *
50 REM *FRANK OCONNOR *
60 REM *****
70 LET K=1
80 LET PSN=5
90 GOSUB 2260
100 LET ZS=0
110 LET SC=0
120 RANDOMIZE
130 LET DRF=0
140 CALL CLEAR
150 PRINT "SKILL LEVEL (1-10)"
160 INPUT "10=EASY 1=HARD" : SL
170 IF (SL>10)+(SL<1) THEN 140
180 CALL CLEAR
190 LET ZS=ZS+1
200 LET DRF=DRF+20
210 CALL SCREEN(7)
220 CALL CHAR(126,"FFFFFFFFFFFFFFFF")
230 CALL COLOR(13,5,1)
240 LET IJN=IJN+1
250 CALL CHAR(159,"1818181800000000")
260 CALL COLOR(16,16,1)
270 CALL CHAR(144,"3C7E7EFFFFFF7E3C")
280 CALL COLOR(15,8,1)
290 LET ROW=12
300 LET CLM=16
310 LET SND=110
320 CALL CHAR(33,"3C3C995A3C183CC3")
330 CALL CHAR(34,"3C3C18FF1899FF")
340 CALL COLOR(1,2,1)
350 REM **SET UP SCREEN**
360 FOR C=1 TO 24 STEP 3
370 CALL HCHAR(C,1,126,32)
380 CALL SOUND(100,323,0)
390 NEXT C
400 FOR C=1 TO 32 STEP 3
410 CALL VCHAR(1,C,126,24)
420 CALL SOUND(100,232,0)

```

```

430 NEXT C
440 LET M$="GAME"
450 LET K=1
460 LET PSN=17
470 GOSUB 2980
480 LET M$=STR$(ZS)
490 LET PSN=21
500 GOSUB 2980
510 LET M$=U$
520 LET K=24
530 LET PSN=1
540 GOSUB 2980
550 LET M$="**HIGH SCORE="
560 LET K=24
570 LET PSN=16
580 GOSUB 2980
590 LET M$=STR$(HS)
600 LET PSN=30
610 GOSUB 2980
620 FOR C=1 TO (SL+28)
630 LET RRR=INT(23*RND)+1

```

```

640 IF RRR=1 THEN 630
650 LET CCC=INT(32*RND)+1
660 IF RRR=12+CCC=16 THEN 630
670 CALL GCHAR(RRR,CCC,G)
680 IF G=32 THEN 630
690 CALL HCHAR(RRR,CCC,144)
700 CALL SOUND(100,789,0)
710 NEXT C
720 CALL HCHAR(ROW,CLM,33)
730 CALL SOUND(100,1000,0)
740 LET M$="ELECTRON"
750 LET PSN=24
760 LET K=1
770 GOSUB 2980

```

How it works

10-350 initialise, define characters, set up variables
360-430 set up grid
440-510 define headings
620-730 print characters
740-850 define headings
860-920 scan keyboard for input
930-1400 move man. Check for losing routine condition.
Sound Routine
1410-1500 place random static on grid
1510-1650 losing subroutine (off grid)
1660-1700 check for high score
1710-1870 ask player for another game and key input
1880-2060 losing subroutine (Hits static)
2070-2220 score update
2230-2490 master card and input for instructions
2510-2960 print instructions
2970-3050 "print at" subroutine
3080-3230 new high score subroutine

Hints on conversion

Texas BASIC has some unusual commands for graphics and sound.

CALL CHAR (number, hexadecimal string) controls the user definition of characters. Number refers to the ASCII code of the redefined character.

CALL HCHAR (row, column, number) prints the character whose ASCII code is number at position row, column on the screen. The TI-99/4A has a screen size of 24 rows by 32 columns.

CALL GCHAR (row, column, variable) is the equivalent of PEEK. The ASCII code of the character at position row, column on the screen is left in the designated variable.

CALL SOUND (d, f, v) sounds a note of frequency, f, duration d milli-seconds and volume v.

CALL KEY (O, K, S) the equivalent of INKEYS, will leave the ASCII code of the key pressed in the variable.

CALL CLEAR clears the screen. A direct equivalent of CLS.

TI-99/4A PROGRAM

```

780 LET M$="YOUR SCORE= "
790 LET PSN=1
800 LET K=1
810 GOSUB 2980
820 LET M$=STR$(SC)
830 LET PSN=12
840 LET K=1
850 GOSUB 2980
860 CALL KEY(3,K,S)
870 IF S=0 THEN 860
880 IF K=68 THEN 930
890 IF K=88 THEN 1020
900 IF K=69 THEN 1110
910 IF K=83 THEN 1200
920 GOTO 860
930 REM MOVE RIGHT#
940 LET CLM=CLM+1
950 IF CLM>32 THEN 1000
960 CALL HCHAR(ROW,CLM-1,128)
970 GOSUB 1290
980 GOSUB 1420
990 GOTO 860
1000 LET CLM=CLM-1
1010 GOTO 860
1020 REM MOVE DOWN#
1030 LET ROW=ROW+1
1040 IF ROW>23 THEN 1090
1050 CALL HCHAR(ROW-1,CLM,128)
1060 GOSUB 1290
1070 GOSUB 1420
1080 GOTO 860
1090 LET ROW=ROW-1
1100 GOTO 860
1110 REM MOVE UP#
1120 LET ROW=ROW-1
1130 IF ROW<2 THEN 1180
1140 CALL HCHAR(ROW+1,CLM,128)
1150 GOSUB 1290
1160 GOSUB 1420
1170 GOTO 860
1180 LET ROW=ROW+1
1190 GOTO 860
1200 REM MOVE LEFT#
1210 LET CLM=CLM-1
1220 IF CLM<1 THEN 1270
1230 CALL HCHAR(ROW,CLM+1,128)
1240 GOSUB 1290
1250 GOSUB 1420
1260 GOTO 860
1270 LET CLM=CLM+1
1280 GOTO 860
1290 REM MOVEMENT+SOUND+CHECKS
1300 CALL GCHAR(ROW,CLM,G)
1310 IF G=32 THEN 1520
1320 IF G=159 THEN 1880
1330 IF G=144 THEN 2080
1340 CALL HCHAR(ROW,CLM,34)
1350 CALL SOUND(100,-3,0)
1360 CALL SOUND(50,-3,15)
1370 CALL HCHAR(ROW,CLM,33)
1380 CALL SOUND(100,2000,3)
1390 CALL SOUND(-30,2000,0)
1400 RETURN
1410 REM STATIC
1420 LET OOO=INT(23*RND)+1
1430 LET PPP=INT(32*RND)+1
1440 IF OOO=1 THEN 1420
1450 CALL GCHAR(OOO,PPP,G)
1460 IF G=32 THEN 1420

```

```

1470 IF G=144 THEN 1420
1480 CALL HCHAR(OOO,PPP,159)
1490 CALL SOUND(10,-3,0,110,0)
1500 RETURN
1510 REM HIT PHOTON (GS)
1520 FOR C=1 TO 10
1530 CALL COLOR(13,16,1)
1540 CALL SOUND(100,-7,0)
1550 CALL COLOR(13,5,1)
1560 CALL SOUND(100,-5,0)
1570 NEXT C
1580 LET M$="YOU FELL INTO A PHOTON"
1590 LET PSN=5
1600 LET K=11
1610 GOSUB 2980
1620 LET M$="GENERATING MODULE....."
1630 LET PSN=5
1640 LET K=12
1650 GOSUB 2980
1660 REM HIGH SCORE CHECK
1670 IF IJN=1 THEN 1680 ELSE 1700
1680 LET HS=SC
1690 GOTO 3060
1700 IF SC>HS THEN 3060
1710 REM END ROUTINE
1720 LET M$="ANOTHER GAME ?"
1730 LET K=18
1740 LET PSN=5
1750 GOSUB 2980
1760 LET M$=I$
1770 LET K=18
1780 LET PSN=18
1790 GOSUB 2980
1800 LET M$="IF YES KEEP Y PRESSED.."
1810 LET K=19
1820 LET PSN=5
1830 GOSUB 2980
1840 CALL KEY(3,K,S)
1850 IF S=0 THEN 1800
1860 IF K=89 THEN 100
1870 END
1880 REM HIT STATIC
1890 LET CLR=1
1890 LET CLR=1
1900 FOR C=1 TO 14
1910 LET CLR=CLR+1
1920 CALL COLOR(1,CLR,CLR+1)
1930 CALL SOUND(100,-7,5,800,0)
1940 CALL SOUND(300,-7,0,800,5)
1950 NEXT C
1960 CALL-COLOR(1,1,1)
1970 CALL SOUND(100,220,0)
1980 LET M$="YOU CAUSED A STATIC"
1990 LET PSN=6
2000 LET K=11
2010 GOSUB 2980
2020 LET M$="OVERLOAD^^^^^^^^^^^^^^"
2030 LET PSN=6
2040 LET K=12
2050 GOSUB 2980
2060 GOTO 1670
2070 REM COLLECT SLIME
2080 FOR C=1 TO 5
2090 LET SND=SND+300
2100 CALL SOUND(10,SND,0)
2110 NEXT C
2120 CALL HCHAR(ROW,CLM,33)
2130 CALL HCHAR(ROW,CLM,34)
2140 LET SC=SC+1

```


TI-99/4A PROGRAM

```

2150 IF SC=DRF THEN 190
2160 LET M$=STR$(SC)
2170 LET PSN=12
2180 LET K=1
2190 GOSUB 2980
2200 IF SND<6000 THEN 860
2210 LET SND=110
2220 GOTO 860
2230 REM MASTER CARD
2240 LET PSN=5
2250 LET K=5
2260 FOR G=1 TO 3
2270 LET M$="!!#ELECTRON#!!"
2280 GOSUB 2980
2290 CALL COLOR(16,16,1)
2300 CALL COLOR(15,8,1)
2310 LET K=K+1
2320 NEXT G
2330 LET PSN=5
2340 LET K=16
2350 LET M$="COPYRIGHT FRANK O'CONNOR..."
2360 GOSUB 2980
2370 LET M$="DO YOU WANT INSTRUCTIONS"
2380 LET PSN=5
2390 LET K=20
2400 REM INSTRUCTIONS
2410 GOSUB 2980
2420 LET M$="Y OR N...."
2430 LET K=22
2440 LET PSN=5
2450 GOSUB 2980
2460 CALL KEY(0,K,S)
2470 IF K=89 THEN 2500
2480 IF K=78 THEN 2490 ELSE 2420
2490 RETURN
2500 CALL CLEAR
2510 PRINT "YOU ARE DEEP INSIDE "
2520 PRINT "THE COUNTRY'S NEWEST"
2530 DISPLAY "THERMO-NUCLEAR PHOTONIAL"
2540 PRINT "P.PH GENERATING PLANT"
2550 PRINT
2560 PRINT "YOUR JOB IS TO MANOUVER"
2570 PRINT "AROUND THE ELECTRO GRID"
2580 PRINT "COLLECTING ALL THE"
2590 PRINT "EXCESS ATOMIC SLIME WHICH"
2600 PRINT "GATHERS ON THE GRID"
2610 PRINT "IN LITTLE BLUE PILES."
2620 PRINT
2630 PRINT "ALSO COLLECTING ON THE GRID"
2640 PRINT "ARE HIGH VOLTAGE STATIC"
2650 PRINT "PARTICLES WHICH MUST BE"
2660 PRINT "AVOIDED AT ALL COSTS"
2670 REM T "AND ARE WHITE."
2680 PRINT
2690 PRINT "PRESS ANY KEY FOR MORE....."
2700 CALL KEY(3,K,S)
2710 IF S=0 THEN 2700
2720 CALL CLEAR
2730 PRINT "POINTS TO REMEMBER ARE....."
2740 PRINT
2750 PRINT
2760 PRINT
2770 PRINT "1.YOU CONTROL THE MAN"
2780 PRINT "WITH THE CURSOR KEYS"
2790 PRINT
2800 PRINT "2.THE GRID IS 100 FT. ABOVE"
2810 PRINT "THE PHOTON GENERATING UNITS"
2820 PRINT "SO DON'T RUN OFF"

```

```

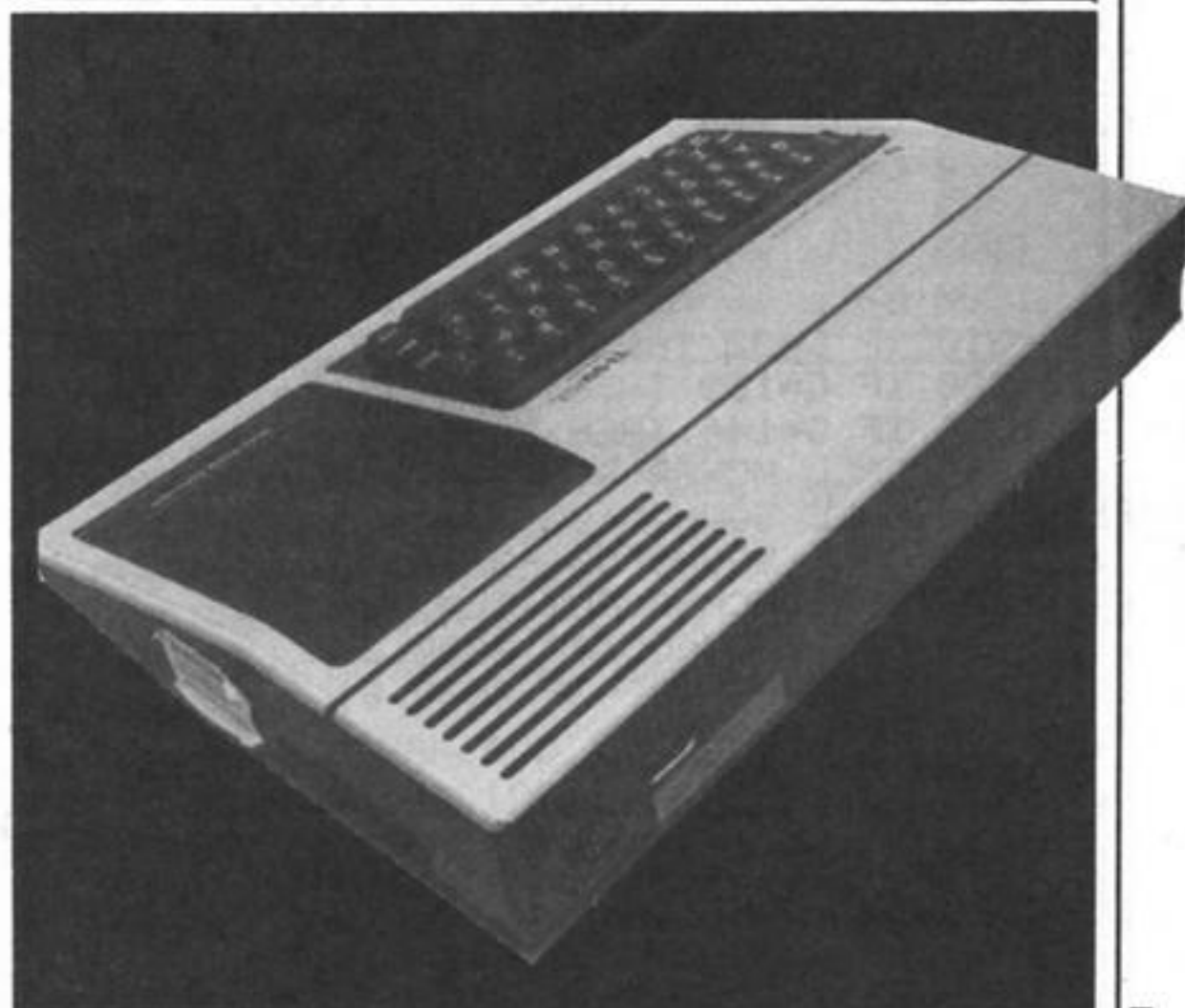
2830 PRINT
2840 PRINT "3.POINTS ARE GIVEN FOR"
2850 PRINT "EACH PILE OF SLIME YOU "
2860 PRINT "COLLECT"
2870 PRINT
2880 PRINT "4.YOU GET A NEW GAME EVERY"
2890 PRINT "20 POINTS."
2900 PRINT
2910 PRINT "PRESS R TO REPEAT"
2920 PRINT "OR P TO PLAY"
2930 CALL KEY(3,K,S)
2940 IF S=0 THEN 2930
2950 IF K=82 THEN 2500
2960 GOTO 140
2970 REM PRINT SUBROUTINE
2980 FOR C=1 TO LEN(M$)
2990 CH=ASC(SEG$(M$,C,1))
3000 CALL HCHAR(K,PSN+C,30)
3010 CALL SOUND(100,3000,0)
3020 CALL HCHAR(K,PSN+C,32)
3030 CALL HCHAR(K,PSN+C,CH)
3040 NEXT C
3050 RETURN

```

```

3060 REM NEW HIGH SCORE
3070 CALL SCREEN(12)
3080 CALL COLOR(1,7,7)
3090 CALL SOUND(1000,262,0,330,0)
3100 CALL SOUND(1000,262,0,392,0)
3110 CALL SOUND(1000,330,0,392,0)
3120 LET M$="A NEW HIGH SCORE***"
3130 LET K=12
3140 LET P=5
3150 GOSUB 2980
3160 LET HS=SC
3170 CALL CLEAR
3180 PRINT "WHAT IS YOUR NAME ? "
3190 CALL COLOR(1,12,12)
3200 INPUT "UP TO 14 LETTERS" U$
3210 IF LEN(U$)>14 THEN 3200
3220 CALL CLEAR
3230 GOTO 1720

```



Double your height VIC-20

Not many VIC-20 owners realise that double height characters can be produced on their machine. But it is possible — just enter the following program:

```
10 POKE 56,24: POKE 55,0
20 A = 6144
30 POKE 36879,27
40 FOR X=A TO 7678 STEP 2
50 B=PEEK (32768+(X-A)
/2)
```

```
60 POKE X,B
70 POKE X+1,B
80 NEXT X
90 POKE 36867,PEEK (36867)
100 POKE 36869,254
110 POKE 36881,20
120 PRINT "(CLR)":NEW
```

Line 120 is to clear the screen. If the program is run/stop and restored, it must be entered again. Make sure the program is saved before running.

Philip Bidwell

Data speed VIC-20 / Commodore 64

This program may be used on a VIC-20 or Commodore 64, speeding up a file saving on both machines. As data is saved at 300 baud, any improvement is a blessing, saving both time and tape space.

The program uses routines in the ROM to save at the same baud rate, but achieves greater speed (usually 10 times faster) by saving individual bytes instead of whole variables, totally in machine code. Thus, saving speed is increased but reliability is exactly the same.

For example, to save a screen to tape, the quickest method in Basic is to use integer variables, which as a file take up five bytes each. The whole screen takes seven minutes to save. Using Dataspeed, variables are stored as 1 byte and the screen takes 40 seconds to save.

To use the program, AS must first be set to the function you want the routine to perform: saving or loading. A% is then set to the start of memory to be saved, and B% to the finish of the memory section. The program may then be run.

Note that on loading, the variables will be stored at the same locations that they were saved from.

Machine code is stored at the top of Basic RAM. To protect it from being erased by variables, the pointer to the top of Basic

Here's a selection of bright ideas sent in by our readers. If you've made a discovery about your micro, why not share it with other enthusiasts by sending it in to us at: Micro Tips, Home Computing Weekly, 145 Charing Cross Road, London WC2H 0EE.

RAM is lowered using lines 10 and 20.

Lines 30 to 60 poke the routine into this RAM. The mid part of the program operates this routine and converts the integer variables into a form usable by machine code. Locations 251 to 254 are vectors for the machine code. Finally, lines 1000 onwards contain data of the machine code to be poked.

David Rees

```
1 REM* DATASPEED *
2 REM* BY *
3 REM*DAVID REES*
10 POKE51,0:POKE52,159
20 POKE55,0:POKE56,159
30 FORN=0TO41
40 READA
50 POKE40704+N,A
60 NEXT N
80 REM * AS=SAVE OR LOAD *
90 AS="SAVE"
100 REM * AX=START B%=FINISH *
110 AX=4096:B%=4352
120 AX=AXAND255
130 AY=(AX/256)AND255
140 BX=B%AND255
150 BY=(B%/256)AND255
160 IF AS="SAVE" THEN 200
170 IF AS="LOAD" THEN 400
180 END
199 REM* SAVE *
200 POKE251,AX:POKE252,AY
210 POKE253,BX:POKE254,BY
220 SYS40704
230 END
399 REM* LOAD *
400 SYS40726
410 END
1000 DATA162,1,160,255
1010 DATA32,186,255
1020 DATA169,0,32,189,255
1030 DATA169,251,166,253
1040 DATA164,254,32,216,255
1050 DATA96
1060 DATA162,1,160,255
1070 DATA32,186,255
1080 DATA169,0,32,189,255
1090 DATA162,255,160,255
1100 DATA32,213,255,96
```

Take a BREAK ZX81

Every ZX81 owner knows that the BREAK key is situated in the bottom right-hand corner of the keyboard. However, we've found that if you press SHIFT, Z, V and B simultaneously, this will also interrupt a program and produce the D report code.

I also have some useful USR routines. If you type RAND USR 836, the display will freeze until only the BREAK key is engaged.

And here's an alternative to typing NEW to clear your program. If you type RAND USR 0 (followed by NEWLINE), this will act in the same way as the NEW command.

ZX81 owners are probably aware of the POKE numbers in Chapter 28 of the ZX81 manual. However, poking address 16384 can affect the screen display very dramatically. Try POKE 16384,40 or POKE 16384,171.

Andrew Barr & Andrew Medcalf

Cap it all Oric-1

I recently discovered yet another way of turning on or off the caps lock on the Oric.

As you will already know, CHR\$(20) switches caps lock, but caps lock has a toggle effect.

To get around this you can PEEK into the screen location where the caps reminder normally is.

So to switch off caps use:
IF PEEK (#BBA4) = 67 THEN
PRINT CHR\$(20)

And use the same line to switch them back on again.

J.B. Laverick

Rolling titles TI-99/4A

Here is a tip for that nice "rolling titles" effect, for the much-neglected TI-99/4A.

```
1 REM ROLLING TITLES
PROGRAM
2 CALL CLEAR
3 FOR M = 1 TO 12
4 CALL COLOR (M, 6, 1)
5 NEXT M
6 CALL SCREEN (16)
7 REM P. M-G, 1983 FOR
HCW
10 FOR I = 1 TO 10
20 READ AS (I)
30 PRINT ::::
```

```
40 PRINT TAB (13) —
LEN(AS(I))/2; AS(I)
50 NEXT I
60 DATA ANYTHING YOU
LIKE, COULD BE PUT, IN
HERE, JUST ALTER THE
LOOP, INCREMENT TO
SUIT
70 DATA YOURSELF, YOU
COULD EXIT PRO-
GRAM, BY
SUBSTITUTING THE
RESTORE, STATEMENT
WITH CALL KEY ETC,
GOOD COMPUTING
80 RESTORE
90 GOTO 10
```

Peter Maskell—Glynn

Amazing draws Spectrum

Try the following:

```
10 PLOT 100,100 : DRAW 10,
10,44
```

This powerful effect is probably the result of a bug in the Spectrum's ROM, but by careful manipulation we can produce some clever patterns using a small amount of memory. I offer no more examples, since this routine is best learned through practice.

Tim Boone

Direct colours Spectrum

Go into Extended mode (cursor E) after a line number and press 2. The cursor will have turned red. What we have done is altered the paper colour directly. CAPS SHIFT followed by a number 0-7 gives us directly the ink colour as printed above the appropriate key.

Using this method we can also directly employ the BRIGHT, FLASH, and INVERSE commands — see the table on page 115 of the Spectrum manual for the combinations.

The users of this are many and varied. You can highlight REM statements, manipulate colours within strings of graphics and even produce blank listings.

After the copyright message on your amazing program, go into extended mode, press CAPS SHIFT 0, and then 0 again in Extended mode. The listing will appear all black: safe from prying eyes!

Incidentally, each direct colour command takes up only two bytes instead of the eight you would use typing INK X. Enough of my chat — go away and practice.

Tim Boone

Side 1

Side 1

Side 2

Side 2

Program name
Side 1

Program name
Side 1

Side 2

Side 2

Your free inlays — just cut them out and place them inside your cassette boxes. They're ideal for recording your own programs and the listings you type in from Home Computing Weekly. Drawings by Chris Adam-Smith

You'll find out just how dangerous mining can be when you play my game for the unexpanded TI-99/4A.

To reach the jewels you must avoid falling rocks and hammers, open mine shafts...and the Mine Guardians. And you have to reach the top before the timer counts down to zero.

After the instructions are displayed there will be a pause while the array holding the positions of the rocks is built up.

Then the screen will clear and the course will be displayed, consisting of a counter in the bottom left, a cave containing the three jewels in the bottom centre and, a passageway across the bottom of the screen with a Mine Guardian at each end.

In the top right will be an open space containing a man and a number telling you whether he is miner 1, 2 or 3.

You move your man using the arrow keys, plus those between them for diagonal movement. He collects a jewel by

Dig for a fortune in gems

Glittering prizes await you in Jewel Mining, a game for the TI-99/4A by James Wilson. But first you must avoid the hazards...and beat the clock

dropping onto it and must then return to the open space above.

Whatever the result, the course is reset and you may then try for the next jewel.

If your man successfully recovers all three jewels, you will be given a score and asked if you wish to play again. You will be given the same option if your man fails to survive but, of course, there will be no score.

The game can be made easier or more difficult by changing certain values:

390 if the numeric value 1.16 is increased more rocks will appear, up to a maximum of 15 per line

570 if the limit value of the FOR-NEXT loop is increased more men will be available.

1060-1080 the ASCII codes for

the counter numbers could be changed to give more or less time

2010 if the numeric value 1.14 is decreased the lasers are less likely to fire and the opposite is also true

2110 changing the figure 1.16 will increase or decrease the likelihood of hammers dropping

Here are a couple of points of interest about the program.

TI BASIC does not support AND/OR statements. However, these can be initiated by enclosing the relational expressions in brackets and using the plus sign for OR and the multiplication sign for AND, as in line 1500 (AND) or in line 1480 (OR).

In lines 1300 and 1310 the new values of I and J are calculated thus: the expressions

Variables

In the interest of saving memory, several variables are used for different purposes.

N controls several FOR...NEXT loops, return variable for GCHAR statements, to detect mine shafts, walls and in timer operating loop

Z is used in several FOR...NEXT loops, counter in building array holding rocks and in displaying rocks, row number in DISPLAY AT sub-routine

HT holds best (lowest) score

I/J row/column position when displaying rocks or new row/column position when moving man

Y/X old row/column position when moving man

MEN number of men

Q start column location in DISPLAY AT sub-routine

JE number of jewels "in transit"

RK return variable in GCHAR statement to detect rock above man's new position

MAN holds ASCII code of man moving left or right

TT score in current game

KK holds ASCII code of key pressed

SS status variable, detects whether key was pressed in CALL KEY statement

P holds ASCII code of space man will leave behind

RV/RH vertical/horizontal locations of a rock above man's new position

WS string variable to hold character code in READ statement, or holds message in DISPLAY AT sub-routine

R(316), S(6,3), G(3,2) arrays holding positions of rocks, mine shafts, jewels

```
89 REM JEWEL MINING
90 REM BY JAMES WILSON
91 REM
100 CALL SCREEN(2)
110 PRINT TAB(9); "JEWEL MINING"; TAB(8); "-----"
120 PRINT " YOU HAVE THREE MEN TO DIG" DOWN TO THE CAVE BELOW
130 PRINT " AND RETURN THE THREE" GEMS, ONE AT A TIME
140 PRINT TAB(8); "TO THE SURFACE" BEWARE OF FALLING ROCKS AND
150 PRINT " HAMMERS, THE MINE GUARDIANS" AND THE OPEN MINE SHAFTS
160 PRINT " DON'T LET THE TIMER RUN" TO ZERO OR YOU WILL DIE
170 PRINT " USE KEYS W,E,R,S,D,Z,X,C TO" MOVE THE MAN LEFT, RIGHT,
180 PRINT " UP, DOWN AND DIAGONALLY" HANG ON FOR ONE MINUTE
```

```
190 FOR N=1 TO 9
200 CALL COLOR(N,10,2)
210 NEXT N
220 FOR N=1 TO 12
230 READ Z,W
240 CALL CHAR(Z,W)
250 NEXT N
260 CALL COLOR(13,15,2)
270 CALL COLOR(14,7,2)
280 CALL COLOR(15,14,15)
290 CALL COLOR(16,4,2)
300 DIM R(316),S(6,3),G(3,2)
310 HT=0
320 GOTO 350
330 CALL CLEAR
340 GOSUB 2830
350 RANDOMIZE
360 Z=1
370 FOR I=1 TO 21
380 FOR J=2 TO 32
390 IF RND#1.16<1 THEN 430
400 R(Z)=J+(I-1)*32
410 Z=Z+1
420 IF Z>1#15 THEN 440
430 NEXT J
440 NEXT I
450 R(316)=Z
460 FOR Z=1 TO 6
470 S(Z,1)=INT(RND#14)+1
480 S(Z,2)=INT(RND#25)+3
490 S(Z,3)=INT(RND#5)+5
500 NEXT Z
510 FOR Z=1 TO 3
520 G(Z,1)=(Z#2)+10
530 G(Z,2)=24
```

```
540 NEXT Z
550 TT=0
560 CALL CLEAR
570 FOR MEN=1 TO 3
580 IF MEN=1 THEN 610
590 Z=24
600 GOSUB 2830
610 CALL HCHAR(1,1,130,784)
620 CALL VCHAR(1,1,32,24)
630 CALL HCHAR(23,3,133,29)
640 Z=1
650 FOR I=1 TO 21
660 IF Z=R(316) THEN 730
670 J=R(Z)-(I-1)*32
680 IF J>32 THEN 720
690 CALL HCHAR(I,J,136)
700 Z=Z+1
710 GOTO 660
720 NEXT I
730 FOR Z=1 TO 6
740 CALL VCHAR(S(Z,1),S(Z,2),32,S(Z,3))
750 NEXT Z
760 CALL VCHAR(6,32,32,9)
770 CALL VCHAR(1,21,32,10)
780 CALL VCHAR(12,10,32,10)
790 FOR I=1 TO 4
800 CALL HCHAR(I,1+25,32,33-(25+I))
810 NEXT I
820 CALL HCHAR(20,12,132,5)
830 CALL HCHAR(21,12,153,5)
840 CALL HCHAR(22,12,32,5)
850 CALL HCHAR(23,12,32,5)
860 CALL VCHAR(20,11,132,4)
870 CALL VCHAR(20,17,132,3)
880 CALL HCHAR(22,10,132)
```

in brackets — for example (KK=69 — will be given a value of zero if they are false or a value of minus one if they are true.

Consequently, adding all these values together, then adding them to the values of Y and X gives new values for I and J.

Hints on conversion

Conversion should not be too difficult on any home computer allowing user-defined characters. Here are explanations of some commands in TI BASIC, all prefaced by the word CALL:

CLEAR equivalent to CLS, places space character in all screen positions

SCREEN changes screen colour according to colour code

CHAR re-defines the standard ASCII characters according to a 16-place hexadecimal string code

COLOR sets colour of the character group in the format (character group, foreground colour, background colour)

HCHAR positions character and optionally repeats it horizontally in the format (row-number, column-number, ASCII code, number of repetitions)

VCHAR as for HCHAR, but repetition is vertically

GCHAR takes the format (row-number, column-number, return variable) and places the ASCII code of the character found in the specified position in the return variable

KEY places ASCII code of key pressed in the key variable, takes the format (key-unit, key variable, status variable) where status variable indicates whether or not a key was pressed

SOUND takes the format (duration, frequency, volume) where the duration is in milliseconds, frequency ranges from 110 to 44733 Hz, volume is 0-30. If a -ve frequency (-1 to -8) is specified, a "white noise" is produced; if a -ve duration is specified, an immediate sound update is carried out when the next CALL SOUND statement is encountered. Optionally, up to three frequencies and volumes may be specified in one CALL SOUND statement to produce chords.

The TI-99/4A uses a screen display of 24 vertical by 32 horizontal in the graphics mode, and 24x28 when printing words. Each square is further divided into an 8x8 matrix which is used for defining the characters

(KK=69) — will be given a value of zero if they are false or a value of minus one if they are true.

TI-99/4A PROGRAM

```

890 CALL HCHAR(22,18,132)
900 CALL HCHAR(24,2,132,31)
910 CALL HCHAR(23,2,138)
920 CALL HCHAR(23,32,137)
930 FOR Z=1 TO 3
940 CALL HCHAR(G(Z,2),G(Z,1),139)
950 NEXT Z
960 W$="MAN "&STR$(MEN)
970 Z=1
980 G=25
990 GOSUB 2850
1000 JE=0
1010 RK=0
1020 MAN=120
1030 X=31
1040 Y=4
1050 CALL HCHAR(4,31,MAN)
1060 CALL HCHAR(24,3,49)
1070 CALL HCHAR(24,4,53)
1080 CALL HCHAR(24,5,48)
1090 FOR Z=1 TO 3
1100 CALL KEY(0,KK,SS)
1110 TT=TT+1
1120 CALL GCHAR(24,5,N)
1130 IF N=48 THEN 1160
1140 CALL HCHAR(24,5,N-1)
1150 GOTO 1250
1160 CALL GCHAR(24,4,N)
1170 IF N=48 THEN 1210
1180 CALL HCHAR(24,5,57)
1190 CALL HCHAR(24,4,N-1)
1200 GOTO 1250
1210 CALL GCHAR(24,3,N)
1220 IF N=48 THEN 1270
1230 CALL HCHAR(24,4,57,2)
1240 CALL HCHAR(24,3,N-1)
1250 IF SS<>0 THEN 1290

```

```

1350 CALL GCHAR(I,J,N)
1360 IF N<>32 THEN 1480
1370 I=Y
1380 GOTO 1270
1390 P=133
1400 IF J>X THEN 1440
1410 IF MAN=120 THEN 1480
1420 MAN=128
1430 GOTO 1460
1440 IF MAN=129 THEN 1480
1450 MAN=129
1460 CALL HCHAR(Y,X,MAN)
1470 GOTO 1270
1480 IF (I<1)+(I>24)+(J<2)+(J>32) THEN 1270
1490 CALL GCHAR(I,J,N)
1500 IF (N<>132)+(N<>136)+(N<>137)+(N<>138) THEN 1550
1510 CALL HCHAR(I,J,144)
1520 CALL SOUND(50,-7,0)
1530 CALL HCHAR(I,J,N)
1540 GOTO 1270
1550 IF I>22 THEN 1580
1560 CALL HCHAR(I,J,144)
1570 CALL SOUND(50,-7,0)
1580 CALL HCHAR(I,J,MAN)
1590 CALL HCHAR(Y,X,P)
1600 Y=I
1610 X=J
1620 IF RK=136 THEN 1840
1630 IF Y>22 THEN 2000
1640 IF (Y>4)+(JE=0) THEN 1660
1650 IF X=24 THEN 2370
1660 CALL GCHAR(I+1,J,N)
1670 IF (N<>32)+(KK=63) THEN 1780
1680 Z=1760
1690 I=I+1
1700 CALL HCHAR(I,J,MAN)
1710 CALL HCHAR(Y,X,32)
1720 CALL SOUND(-500,Z,2)

```

```

1260 NEXT Z
1270 IF RK=136 THEN 1840
1280 IF Y>22 THEN 2000 ELSE 1090
1290 P=32
1300 I=Y+(KK=69)-(KK=88)-(KK=67)-(KK=98)+(KK=87)+(KK=82)
1310 J=X+(KK=83)-(KK=68)-(KK=67)-(KK=82)+(KK=98)+(KK=87)
1320 IF J<>X THEN 1390
1330 IF I=Y THEN 1270
1340 IF (KK=69)+(I=0)+(Y=24) THEN 1480

```

How it works

100-210 set screen color (black), print instructions and sets upper case letter colors
 220-250 define characters from DATA statements in lines 2890, 2900, 2910
 260-290 set colors of characters
 300 DIMensions arrays
 330-340 clear screen, print "wait" message for second and subsequent games
 350 seeds random number generator
 360-450 set array values holding random positions of rocks
 460-500 set array values holding position, depth of shafts
 510-540 set array values holding positions of jewels
 550-560 clear screen, set current game score to zero
 570 FOR-NEXT loop controlling number of men
 590-600 display wait message
 610-950 display course
 960-990 display number of current men
 1000-1040 set game variables to start positions
 1050-1080 display man and counter at start positions
 1090 control loop for delay on key press
 1100 detects key pressed in variable KK
 1110 adds 1 to score
 1120-1240 operate on-screen counter
 1250 detects whether key was pressed
 1270-1280 check for rock detected in previous run through or for man in cave
 1290&1390 set value of P depending on key pressed
 1300-1310 set new values of I and J
 1320-1330 check if incorrect key pressed

1340-1380 check to see if an open space is above old position if the man is moving up and stops the man if there is
 1390-1470 change direction of man
 1480 checks to see if man is at screen edge
 1490-1540 checks for rocks and walls in new position
 1550-1610 move man
 1620 checks if rock is set to fall
 1630 checks if man is in cave
 1640-1650 checks to see if man has returned with jewel
 1660-1770 checks for open mine shaft below man and drops man down shaft
 1780-1830 check for rock above new position
 1840-1990 falling rock routine
 2000 checks to see if man is in cave or passageway
 2010-2070 laser fire routine
 2080-2100 opens cave exit, closes cave entrance
 2110-2230 falling hammer routine
 2240-2260 checks to see which jewel is picked up
 2270-2310 dead man routine
 2320-2360 all men dead message
 2370-2460 returned jewel routine, checks to see if all three jewels returned
 2470-2360 messages for a win, compares current score with best score, displays best and current score
 2640-2720 messages to decide whether to play again
 2730-2820 CALL KEY routine to select whether to play again
 2830-2880 DISPLAY AT sub-routine
 2890-2910 DATA statements containing codes for defining characters



```

1730 CALL GCHAR(I+1,J,N)
1740 IF (N<>32)+(N<>133) THEN 2270
1750 Z=Z-35
1760 Y=I
1770 GOTO 1690
1780 IF I=1 THEN 1090
1790 CALL GCHAR(I-1,J,RK)
1800 IF RK<>136 THEN 1090
1810 RV=I-1
1820 RH=J
1830 GOTO 1090
1840 RK=RV
1850 CALL GCHAR(RV+1,RH,N)
1860 CALL HCHAR(RV+1,RH,136)
1870 CALL HCHAR(RV,RH,32)
1880 CALL SOUND(-1000,-5,2)
1890 IF RV=1 THEN 1940
1900 CALL GCHAR(RV-1,RH,Z)
1910 IF Z<>136 THEN 1940
1920 RV=RV-1
1930 GOTO 1860
1940 IF N=MAN THEN 2270
1950 CALL GCHAR(RK+2,RH,N)
1960 IF (N<>32)+(N<>133)+(N<>MAN) THEN 1630
1970 RV=RK+1
1980 CALL SOUND(-1000,-6,0)
1990 GOTO 1840
2000 IF (J>11)+(J<17) THEN 2080
2010 IF RND<1,14<1 THEN 1090
2020 CALL SOUND(500,800,2)
2030 IF J<12 THEN 2060
2040 CALL HCHAR(23,J,152,32-J)
2050 GOTO 2270
2060 CALL HCHAR(23,3,152,J-2)
2070 GOTO 2270
2080 IF JE<>0 THEN 2110
2090 CALL HCHAR(23,17,132)
2100 CALL HCHAR(23,11,133)
2110 IF RND<1,16<1 THEN 2240
2120 Q=INT(RND*5)+12
2130 FOR Z=22 TO 24
2140 CALL GCHAR(Z,Q,N)
2150 IF (N=139)+(N=132) THEN 2210
2160 CALL HCHAR(Z,Q,153)

```

```

2170 CALL HCHAR(Z-1,Q,32)
2180 CALL SOUND(-200,INT(10000/Z),2)
2190 IF N=MAN THEN 2270
2200 NEXT Z
2210 CALL SOUND(60,-7,0)
2220 CALL HCHAR(Z-1,Q,32)
2230 CALL HCHAR(21,Q,153)
2240 IF Y<>24 THEN 1090
2250 JE=(X-10)/2
2260 GOTO 1090
2270 CALL SOUND(-500,-5,0)
2280 W$="YOUR MAN'S DEAD"
2290 Z=12
2300 GOSUB 2840
2310 NEXT MEN
2320 CALL SOUND(1500,110,3,220,2,330,1)
2330 W$="ALL YOUR MEN ARE DEAD"
2340 Z=10
2350 GOSUB 2840
2360 GOTO 2640
2370 G(JE,1)=32
2380 G(JE,2)=JE
2390 FOR Z=1 TO 11
2400 CALL SOUND(40,1000,3)
2410 NEXT Z
2420 W$="JEWEL "&STR$(JE)&" RETURNED"
2430 GOSUB 2840
2440 FOR I=1 TO 3
2450 IF G(I,1)<>32 THEN 590
2460 NEXT I
2470 CALL SOUND(200,370,4,550,4,294,4)
2480 CALL SOUND(400,490,3,550,3,350,3)
2490 CALL SOUND(200,490,2,550,2,350,2)
2500 CALL SOUND(600,390,1,550,1,470,1)
2510 IF HT<>0 THEN 2540
2520 HT=TT
2530 GOTO 2550
2540 IF HT>TT THEN 2520
2550 W$="WELL DONE,"
2560 Z=8
2570 GOSUB 2840
2580 W$="YOU'VE RETURNED ALL 3 JEWELS"
2590 Z=10
2600 GOSUB 2840

```

```

2610 W$="YOUR SCORE "&STR$(TT)&" BEST SCORE "&STR$(HT)
2620 Z=12
2630 GOSUB 2840
2640 W$="DO YOU WANT TO PLAY AGAIN"
2650 Z=Z+2
2660 GOSUB 2840
2670 W$="PRESS S FOR SAME SET-UP,N FOR"
2680 Z=Z+2
2690 GOSUB 2840
2700 W$="NEW SET-UP OR F TO FINISH"
2710 Z=Z+2
2720 GOSUB 2840
2730 Z=12
2740 CALL KEY(0,KK,SS)
2750 IF SS=0 THEN 2740
2760 ON POS("SNF",CHR$(KK),1)+1 GOTO 2740,510,330,2770
2770 CALL CLEAR
2780 W$="THANKS FOR PLAYING"
2790 GOSUB 2840
2800 FOR Z=1 TO 1500
2810 NEXT Z
2820 STOP
2830 W$="HANG ON FOR 1 MINUTE"
2840 Q=16-INT(LEN(W$)/2)
2850 FOR N=1 TO LEN(W$)
2860 CALL HCHAR(Z,Q+N,ASC(SEG$(W$,N)))
2870 NEXT N
2880 RETURN
2890 DATA 128,0C0C3C0C1C14340C,129,30303C3030182C30,130,AA55AA55AA55AA55,132,FFF
2900 DATA 133,0000000000000055,136,1030707C3E3C7C30,137,0303FF000F03030F,138,C0C
2910 DATA 139,00000010307C3010,144,55AA55AA55AA55AA,152,0000FF,153,10101030

```


Invasion TI-99/4A £5

Stainless Software, 10 Alstone Road, Stockport, Cheshire, SK4 5AH

Here you have the task of single-handedly defending your island from landing parties of enemy ships. The display on screen shows the coastline and your gun, which is strategically placed near the bottom in a bay.

The invasion force appears, one at a time, to the left of the screen and travel horizontally towards the shore.

Using keyboard controls to move and fire, your gun is placed so as to blast each ship as it passes. When one is hit, however, it leaves a shipwreck which forms a barrier to any further gunfire.

As the game progresses and more ships are destroyed direct

hits become increasingly difficult. Should a ship land on your island the width of the bay in which your gun is placed becomes narrower. The game ends when either you accidentally crash into the bay or complete invasion is accomplished.

A simple yet enjoyable game, well crashproofed and with some pleasing graphics, but it is one which you can never actually win.

The only goal is to achieve the highest score possible before you are ultimately stopped. Some comment on your performance at the end of each run might well make this more motivating. J.W.

instructions	80%
playability	60%
graphics	80%
value for money	60%



Hen House TI-99/4A £27.95 (cartridge)

Galaxy, 60 High Street, Maidstone, Kent

The player controls, using keyboard or joystick, a farmer

who has two conflicting tasks — to catch eggs falling from a hen house and place them in a truck, while also shooting poachers and foxes (and the odd bird).

Points are scored for each bird or poacher shot, and the game ends when six eggs have been dropped. The pace of the game speeds up as you go along, but there is only one scenario; this

Games to take home to the range

Our reviewers put on their Stetsons and tell you partners about some Texas games on cassette and cartridge

could give the game limited playability.

Instructions are far too brief but adequate if you don't mind a little experimentation. You pick the gun up and drop it by moving to screen left, and put the eggs in the lorry by moving to screen right.

The graphics are quite good, especially the animation of the farmer. S.S.

instructions	40%
playability	50%
graphics	80%
value for money	60%



Torpedo Alley/ Ascot Stakes TI-99/4A (Extended BASIC) £8.00

Lantern Software, 4 Haffenden Road, Tenterden TN30 6QD

Simple spelling mistakes in the instructions for both programs do little to improve them. While excellent use is made of graphics, interest is not sustained, and skill requirements are negligible (or absent in the case of Ascot Stakes).

Torpedo Alley has you launching two torpedoes simultaneously at three ships running the gauntlet between the rocks at night. Only two degrees of difficulty are offered — easy or hard — the difference lying mainly in the game tempo.

The score is displayed as the

'time' counts up to 120. The game might well appeal to the very young or to those with poor hand-eye co-ordination.

Ascot Stakes is graphically excellent, producing a television-type pan of the race, but there is no skill involved at all. Punters select their horses, place their bets (you start with £100) and the computer does the rest.

You play until either all punters are broke or want to give up — you can't stop unless everybody else wants to!

This game is likely to appeal to only the very young. Although providing two programs for the price, the cost is still very high. P.B.

instructions	75%
playability	30%
graphics	90%
value for money	30%



Rabbit Trail TI-99/4A £27.95 (cartridge)

Galaxy, 60 High Street, Maidstone, Kent

The screen shows a rabbit run. You are the bunny — very nice graphics here. You must hop along your rabbit run collecting carrots and avoiding some nasty

creatures that seem to like rabbits...

The screen display seems to be a rabbit warren, but on the basis that hawks don't usually fly underground, I assume it must be meant to be a rabbit run, or a series of runs connected by tunnels.

There are seven screens to try your hand at, and the game is against the clock.

Graphics are good, and the

game scores highly for originality and lack of violence (provided you avoid the predators).

Enjoyable. But owners of the 99/4 should note that this module will only run on the 99/4A! S.S.

instructions	50%
playability	80%
graphics	95%
value for money	80%



Core TI-99/4A £3.50

Christine Computing, 6 Florence Close, Watford, Herts.

Hidden deep in the earth is the core. To reach it you have to complete all three stages of this game. Firstly your craft descends through clouds, birds and aeroplanes and you have to navigate a clear path.

Secondly, after successfully through the sky you go underground negotiating the cave and caverns.

It takes a little while to realise that there is only one way through but if you complete this the third part looms up and all that stands between you and the core is the 3D maze.

After a number of frustrating attempts you should reach the core. A little light hearted relief will be found here.

If you collide with any part of the scenery your score and the highest so far is displayed along with an invite to play again but no score is given for a successful mission.

Each new game is the same as the last, so once you have reached the core a few times there is very little incentive to play again.

I had no problem in loading this game and found the screen instructions clear, although the spelling of the word 'plains' for planes was a little ambiguous.

The graphics are neat but colour would enhance the game. C.E.

instructions	85%
playability	50%
graphics	50%
value for money	50%



Save selectively VIC-20

Owners of the VIC 20 micro are now able to save selective memory blocks on to tape using the steps below. The SAVE command only saves programs.

- 1 POKE locations 193 and 194 with the start address of the memory block.
- 2 POKE locations 174 and 175 with the end address of the memory block.
- 3 POKE locations 186 with the device number (1 for cassette, 8 for disk etc.) and location 185 with 3.
4. POKE location 147 with 0 for load and 1 for verify.

To perform save simply SYS 63109 and to perform load simply SYS 62795 (this routine doesn't inform you when it has found something).

The main uses of saving selective memory blocks are for performing screen text dumps onto tape and also to perform a character generator dump onto tape (hi-res picture saving). Now there is no need to load characters from data statements.

```
1 REM SCREEN DUMP PROG DEMO
2 REM INSERT TEXT TO BE SAVED BETWEEN LINES 20-100
10 PRINT CHR$(147)CHR$(5): POKE 36879,8
20 PRINT "TEXT TO BE SAVED"
100 GOSUB 1000
110 PRINT:PRINT "WANT TO LOAD SCREEN?"
120 GET A$:IF A$ = "N" THEN END
130 IF NOT A$ = "Y" THEN 120
140 GOSUB 2000
150 GOTO 150:REM FREEZE
999 REM SAVE SCREEN ROUTINE
1000 GOSUB 3000
1010 IF PEEK (37148) = 254 GOTO 1010
1020 SYS 63109 : RETURN
1999 REM LOAD SCREEN ROUTINE
2000 PRINT CHR$(147) : GOSUB 3000
2010 SYS 62795 : RETURN
2999 REM SET UP
3000 POKE 193,0 : POKE 194, PEEK(648)
3010 POKE 174,0 : POKE 175, PEEK(648)+2
3020 POKE 186,1 : POKE 185,3
3030 POKE 147,0 : RETURN
```

Line 110 waits for the user to press play and record.

The routines don't display any messages such as saving etc.

When run press play and record.

David Redmonds

Got a bright idea for using your micro? This is the page where you can share your brainwave with other computer enthusiasts. Send your hints, warnings, discoveries and suggestions to Micro Tips, Home Computing Weekly, 145 Charing Cross Road, London WC2H0EE

Find that location VIC-20 plus 3K

When working out user defined graphics etc, it is always a problem to find out the exact memory location to POKE the new values to. This line of program, when used with DATA statements solves this very quickly.

```
LN FOR CH = 0 TO CN:READ C$: C = (ASC(C$) - 64) * 8 = 7168: FOR CS = 0 TO 7: READ CQ: POKE C+CS, CQ: NEXT CS,CH
```

where LN = line number, and CN the number of characters to define.

A matching DATA line would be:—
LN DATA "X",123,234,0,34,62,1,3,0

David Shepherdson

Make an introduction Any micro

Before you load data into your computer do you introduce the program via the TV speaker? With the tape recorder linked through to the TV, you don't have to be an expert to realise that this is possible.

For the interest of beginners like myself, here is how I go about the operation with my TI-99/4A.

Start when you are ready to SAVE your program on to tape. Do not connect computer cable to recorder. Rewind tape to 000 or selected starting number. Record intro in normal manner.

Example: "HCW June 7, '83 program. Subject: Game. Title: Last Stand by Vince Apps. The aliens have invaded the earth and you are the last survivor. In your tank you strive to reach alien headquarters and destroy it before you are wiped out. You can do this by hiding behind trees as you advance... Typed in 27 June...stop recording and ad-

vance meter reading by 4 numbers NOW."

You have finished voice recording. Disconnect mike (if not built-in), connect computer cable. Next start SAVE routine, remembering that you do not need to rewind, but advance four numbers instead.

With your program on cassette, the play procedure is straight-forward. attach computer cable. Start from your beginning number, listen to the recorded message and press the recorder STOP button immediately as instructed. Do not press enter. Advance the four numbers and proceed in normal manner, that is, press enter but do not rewind.

You may find this method a convenient way of storing information. You do not have to listen to the introduction every time but it is there when needed.

To me it seems quicker and more personal than typing in a screened introduction. The cassette file card would read (say) 1-12 Intro, 16-26 "Last Stand".

Joe Berry

Fun with colour TI-99/4A

This short program for the TI-99/4A shows off the colours in a bright display.

```
10 CALL CLEAR
20 CALL SCREEN (15)
30 X=3
40 FOR T=40 TO 144 STEP 8
50 CALL CHAR (T,"FFFFFFFFFFFFFFFF")
60 CALL VCHAR(1,X,T,48)
70 X=X+2
80 NEXT T
90 FOR T=2 TO 15
100 CALL COLOR (T,T,T)
110 NEXT T
120 G=1
130 FOR T=15 TO 2 STEP -1
140 CALL COLOR (T,G,1)
150 G=G+1
160 NEXT T
170 GOTO 90
```

Neil Sherwood

Protect your screen 48K Oric 1

Owners of the 48K ORIC who wish to protect part of the screen from scrolling can alter the value stored at address #26F. The value stored determines the number of lines to be used for normal scrolling printing etc.

Changing the value takes just one simple line:—

```
DOKE #26D, #BB58:POKE #26F,x:CLS
```

The variable x being the value to be stored.

This line could obviously be incorporated in a program if necessary

Colin Failes

Practical POKES VIC-20

On the VIC-20 there are lots of POKES to do weird and wonderful things, but for some reason they are not listed in the user manual. So here I have made a list of some of them. I hope they prove useful to you...

Poke 808,127 disables the RUN/Stop key (any memory VIC)

Poke 808,112 re-enables the RUN/Stop key.

Poke 36864,(0 to 128) moves the screen horizontally.

Poke 36865,(38 onwards) = Moves the screen vertically.

Poke 36866,(1 onwards) = Sets the number of chars on a line.

Poke 36867,(1 onwards) or 128 = Sets the number of lines.

Poke 199,1 = Turns the reverse field on.

Poke 199,0 = Turns the reverse field off.

Poke 204,0 = Flashes the cursor in a GET statement.

Poke 211,(0 to 22) = Sets the cursor column position.

Poke 214,(0 to 22) = Sets the cursor row position.

Poke 649,0 = Disables the keyboard.

Poke 649,10 = Re-enables the keyboard.

Poke 650,255 = Makes every key repeat.

Poke 650,0 = Sets repeat functions to normal.

An interesting effect is obtained by entering:

```
Poke 36879,8
```

```
CTRL 2
```

```
Poke 788,223
```

Then move the cursor around with the cursor keys.

Alan Blackham

Are they worth the money?

Cartridge games reviewed by Mike Roberts. Most also have a joystick option

Parsec £29.95

This is the most beautiful arcade game for the TI that I have ever seen.

The use of the speech synthesiser from within the program is excellent. It simulates an on-board combat computer warning you of impending attack and other warnings such as running out of fuel, and it (or rather she as it is a female voice) will give encouraging remarks as to your progress, a pleasant change from some sarcastic computer games.

The concept of the game is similar to the scramble/defender type, with the space ship travelling horizontally across a moving planetary surface strewn with objects. If you hit this you lose one

of your initial five lives. New ships are granted at regular intervals.

There are eight different types of nasties out to get you and they come in waves, so you have to destroy all of the first type before you go onto the next. If this is not enough for you, you are in constant danger of your laser overheating and your ship blowing up, running out of fuel and having to fly down a tunnel to get some more, and flying through an asteroid belt.

There are 16 different levels to get through, and although I have been playing this game for a week now I have only got to level 4, so it is very challenging.

instructions	100%
playability	100%
graphics	100%
value for money	100%



Alpiner £24.95

A very interesting game with a concept that I haven't seen before. The game needs explaining in some detail as it is different to anything else on the market.

The idea is to climb up six of the world's biggest mountains while avoiding falling rocks, rotten eggs, bats, trees, etc. The speech synthesiser option generates male and female voices which will warn you of impending doom and give encouraging remarks.

There is an option for two players and seven differing languages are available, although the speech synthesiser still speaks in English. Points are awarded for how high you get, plus various other bonus points.

The graphics are excellent, and produce a good picture of a cliff face with all the animals and other dangers well represented, and a very realistic climber that moves properly as he climbs up the mountain.

The instruction book is up to TI's usual standard and runs to 18 pages, and is quite comprehensive.

Overall this game would be good for someone who wants something that is a bit different to the normal run of the mill zap-the-klingsons game.

instructions	100%
playability	85%
graphics	100%
value for money	60%



Car Wars £19.95

The object is to race around a five-lane race track. Of course there is a computer controlled car racing round in the opposite direction to you trying its very best to smash you up.

The lanes of the race track have four points where you can change up to two lanes. The lanes are filled with dots, which you have to run over, Pac-man fashion, to obtain points. After a user defined number of dots have been cleared the car will speed up. The other difficulty setting is a choice of three speeds, 'creepin', fast, and flyin'.

A bonus is awarded from 200 to 1000 points depending on skill level.

The instructions are good, ex-

cept that they don't mention that the arrow keys can be used in place of a joystick. The graphics are quite good and the game is very fast.

The only real gripe against this game is that it is very simple and, after a while, boring. It doesn't warrant the price or to be put in a cartridge. When Commodore has a simple but amusing game like Blitz they put it on tape and don't charge an extortionate amount for it.

But aside from that this is a well put together game with good documentation and instructions.

instructions	100%
playability	60%
graphics	90%
value for money	40%



TI Invaders £19.95

Space invaders is getting a bit long in the tooth and TI Invaders is no exception.

This program is a good representation of the game that took the world by storm so many years ago. There are some differences — aren't there always? — and in this case they are for the better. There are two skill levels, called "merely aggressive" and "downright nasty" which is a nice touch. The instructions are very clear and concise except for the point that the keyboard arrow keys can be used instead of a joystick.

After each screen of invaders you can shoot it out with a mystery ship on its own.

A new laser base is awarded at 3,000 points and at 10,000 points

one of your demolished forts is repaired.

The graphics are good. Although they look like invaders they don't look like the arcade invaders. The laser bases are good and when one is destroyed it turns into a wreck and is lowered into an underground cavern and another will take its place.

When the invaders have ultimately won they will jump up and down with much glee.

This game is getting a bit obsolete and boring, but it's not TI's fault and this is a good game despite the limitations of the game concept.

instructions	80%
playability	40%
graphics	90%
value for money	30%



Video Chess £34.95

This cartridge gives as much as could be reasonably expected from a computer chess program. Unfortunately, it is rather overpriced compared to other chess programs available for other computers.

The instructions first give an explanation of the game of chess itself and the notation used to describe your moves to the computer.

The manual then goes on to explain the various options available to the user.

These are all menu-driven. The first menu has six options: play game, which starts the play; selection of game type, which gives you three other menus — level of play, style of play, or colour; chess board, which enables

two human players to play; set up a problem, this means you can set up the board in any position and to play the computer from there; play more than one game simultaneously, up to nine games can be played; and to load in a game from cassette tape.

When playing a game you can cancel the last move, resign, offer a draw to the computer, replay from start, change sides, rearrange the pieces, change the style of play, correct the last move, and save the current status on cassette tape.

Castling, pawn promotion, and en passant are all supported.

instructions	100%
ease of use	85%
display	100%
value for money	60%



Will you get to the treasure first?

Hidden somewhere on Treasure Island is a chest crammed with gold. Can you find it before the pirate, who is also scouring the island?

You must use your skill and judgement to work out the location of the treasure, using the clues provided.

Then use the arrows keys E S D X (with alpha lock button down) to go and find it.

The clues will tell you how far, in whole squares, the treasure is from the mountains and the jungle. When the treasure doesn't turn out to be where you thought it was, the game can become quite frustrating and addictive.

The pirate is armed to the teeth, so if you meet him you

There are more dangers than just a bloodthirsty pirate in Lance Booth's Treasure Island game for the standard TI-99/4A. Type it in and try for a chestful of doubloons

How it works

120-170 define characters
180 sets screen colour
190-270 set character colours
290 random numbers
300-350 set fixed variables
360-370 check treasure not in jungle
380-390 calculate distances to treasure
400 clears screen
410-510 display clues
590-810 display island and landscape
820 hides treasure
830-840 initial pirate location
850 stores old character at pirate square
860-870 initial player location
880-890 initial pirate and player display
900-910 wait for player to start
920-930 initial pirate direction
950-1260 main game sequence
950 check keyboard
960 clears old player
970-1040 check for movement
1050 checks new square
1070-1130 check for special situations
1140 displays new player
1150 clears old pirate
1160-1170 move pirate position
1180 checks new pirate square, stores old character
1190-1200 check for special situations
1210-1220 check pirate still on island
1230 displays new pirate
1240-1250 sound effects
1260 sends program back to start of game sequence
1270-1310 change pirate direction if needed
1320-1340 fell in sea
1350-1370 fell in river
1380-1400 fell in swamp
1410-1430 lost in jungle
1440-1460 lost in mountains
1470-1490 pirate caught you
1500-1520 pirate found treasure
1530-1550 sound effects
1560-1670 you found gold
1680-1700 display message
1710-1740 play again?
1750 end

won't stand a chance. There are other dangers to watch out for too, so take care.

You can alter the difficulty level in two ways.

First, remove lines 900 and 910. This will cause the game to start as soon as the screen display is complete, rather than waiting for you to initiate play.

Alternatively, the number of swamps can be increased by rais-

ing the value of the FOR...TO statement at line 720. This will mean concentrating on negotiating the swamps as much as on finding the treasure and avoiding the pirate and the other dangers.

When each turn is over — whatever the reason — the location of the treasure is displayed. X marks the spot where it was hidden.

```
100 REM TREASURE ISLAND
110 REM BY LANCE BOOTH
120 CALL CHAR(97,"FFFFFFFFFFFFFFFF")
130 CALL CHAR(96,"00001054307C")
140 CALL CHAR(120,"003C7E7E3C1010")
150 CALL CHAR(136,"0010103C3CFE")
160 CALL CHAR(144,"101030541020206C")
170 CALL CHAR(152,"3C5A7E3C10661066")
180 CALL SCREEN(5)
190 CALL COLOR(2,4,4)
200 FOR I=3 TO 8
210 CALL COLOR(1,16,2)
220 NEXT I
230 CALL COLOR(9,5,4)
240 CALL COLOR(13,13,4)
250 CALL COLOR(14,7,4)
260 CALL COLOR(15,16,4)
270 CALL COLOR(16,2,4)
280 REM SET VARIABLES
290 RANDOMIZE
300 JR=INT(15*RND)+5
310 JC=INT(12*RND)+16
320 MR=INT(17*RND)+4
330 MC=INT(12*RND)+14
340 TR=INT(10*RND)+4
350 TC=INT(17*RND)+14
360 IF (TR<JR-1)+(TR>JR+2) THEN 380
370 IF (TC>JC-2)*(TC<JC+2) THEN 340
380 X=INT(SQR((TC-JC)*(TC-JC)+(TR-JR)*(TR-JR)))
390 Y=INT(SQR((TC-MC)*(TC-MC)+(TR-MR)*(TR-MR)))
400 CALL CLEAR
410 PRINT "TREASURE"
420 PRINT
430 PRINT Y;" FROM"
440 PRINT
450 PRINT "MOUNTAINS"
460 PRINT
470 PRINT X;" FROM"
480 PRINT
490 PRINT "JUNGLE"
500 PRINT
510 PRINT "CAN YOU"
520 PRINT
530 PRINT "FIND IT"
540 PRINT
550 PRINT "BEFORE THE"
560 PRINT
570 PRINT "PIRATE"
580 REM DISPLAY ISLAND
590 FOR R=4 TO 21
600 CALL HCHAR(R,14,43,17)
610 NEXT R
```

Hints on conversion

The program should convert fairly easily to any computer with user-defined characters. Here is an explanation of some of the TI BASIC commands I have used:

CALL CHAR (ASCII code number, hexadecimal string) defines characters used in the program. The hexadecimal string defines the pattern to be given to the character with the ASCII code stated

CALL HCHAR (row number, column number, ASCII code, number of repetitions) places character with ASCII stated, at row and column specified, and optionally repeats it horizontally the number of times stated

CALL VCHAR (row number, column number, ASCII code, number of repetitions) works in the same way as CALL HCHAR only repeats vertically

CALL GCHAR (row number, column number, numeric variable) equivalent of PEEK, assigns ASCII code of character at row and column stated to numeric variable

CALL CLEAR clears the screen

CALL COLOUR (character set, foreground colour code, background colour code) specifies foreground and background colours for all characters in set specified

CALL SCREEN (colour code) specifies screen colour using colour code

CALL SOUND (duration, frequency, volume) produces sound. Duration in milliseconds, frequency in Hertz (or noise code), and volume in scale from 0 loudest to 30 quietest

CALL KEY (key unit,k,s) equivalent of INKEY\$, returns code of key pressed in variable k



TI-99/4A PROGRAM

```

620 CALL VCHAR(15,13,43,5)
630 CALL HCHAR(3,13,43,4)
640 CALL HCHAR(3,23,43,9)
650 CALL VCHAR(2,31,43,11)
660 CALL HCHAR(22,19,43,9)
670 REM DISPLAY LANDSCAPE
680 CALL HCHAR(JR-1,JC-1,128,4)
690 CALL HCHAR(JR,JC-2,128,4)
700 CALL HCHAR(JR+1,JC-2,128,5)
710 CALL HCHAR(JR+2,JC-1,128,4)
720 FOR I=1 TO 10
730 SR=INT(18*RND)+4
740 SC=INT(17*RND)+14
750 CALL GCHAR(SR,SC,GET)
760 IF GET<43 THEN 730
770 CALL HCHAR(SR,SC,96)
780 NEXT I
790 CALL HCHAR(MR,MC,136,5)
800 CALL VCHAR(4,20,97,4)
810 CALL VCHAR(7,19,97,6)
820 CALL HCHAR(TR,TC,44)
830 PR=INT(17*RND)+5
840 PC=INT(15*RND)+15
850 CALL GCHAR(PR,PC,GETP)
860 HR=22
870 HC=23
880 CALL HCHAR(PR,PC,152)
890 CALL HCHAR(HR,HC,144)
900 CALL KEY(3,K,S)
910 IF S=0 THEN 900
920 RDIR=1
930 CDIR=1
940 REM GAME
950 CALL KEY(3,K,S)
960 CALL HCHAR(HR,HC,43)
970 IF K<>83 THEN 990
980 HC=HC-1
990 IF K<>68 THEN 1010
1000 HC=HC+1
1010 IF K<>69 THEN 1030
1020 HR=HR-1
1030 IF K<>88 THEN 1050
1040 HR=HR+1
1050 CALL GCHAR(HR,HC,GETH)
1060 REM POSITION CHECK
1070 IF (GETH=32)+(GETH=31) THEN 1320
1080 IF GETH=97 THEN 1350
1090 IF GETH=96 THEN 1380
1100 IF GETH=128 THEN 1410
1110 IF GETH=136 THEN 1440

```

```

1120 IF GETH=152 THEN 1470
1130 IF GETH=44 THEN 1500
1140 CALL HCHAR(HR,HC,144)
1150 CALL HCHAR(PR,PC,GETP)
1160 PC=PC+CDIR
1170 PR=PR+RDIR
1180 CALL GCHAR(PR,PC,GETP)
1190 IF GETP=44 THEN 1500
1200 IF GETP=144 THEN 1470
1210 IF (PC<14)+(PC>30) THEN 1270
1220 IF (PR<4)+(PR>21) THEN 1300
1230 CALL HCHAR(PR,PC,152)
1240 CALL SOUND(100,110,0)
1250 CALL SOUND(100,220,0)
1260 GOTO 940
1270 CDIR=-CDIR
1280 IF (PR<4)+(PR>21) THEN 1300
1290 GOTO 940
1300 RDIR=-RDIR
1310 GOTO 940
1320 GOSUB 1530
1330 M$="THE SHARKS IN THE SEA ATE YOU"
1340 GOTO 1600
1350 GOSUB 1530
1360 M$="YOU DROWNED IN THE RIVER"
1370 GOTO 1600
1380 GOSUB 1530
1390 M$="YOU FELL IN THE SWAMP"
1400 GOTO 1600
1410 GOSUB 1530
1420 M$="YOU'RE LOST IN THE JUNGLE"
1430 GOTO 1600
1440 GOSUB 1530
1450 M$="YOU'RE LOST IN THE MOUNTAINS"
1460 GOTO 1600
1470 GOSUB 1530
1475 CALL HCHAR(PR,PC,152)
1480 M$="THE PIRATE CAUGHT YOU"
1490 GOTO 1600
1500 GOSUB 1530
1510 M$="THE PIRATE FOUND THE TREASURE"
1520 GOTO 1600
1530 REM FAILURE MUSIC
1540 FOR N=-8 TO -1
1550 CALL SOUND(+350,N,0)
1560 NEXT N
1570 RETURN
1580 REM SUCCESS
1590 RESTORE 1640

```

```

1600 READ TONE,DRN
1610 IF DRN=0 THEN 1670
1620 CALL SOUND(DRN,TONE,0)
1630 GOTO 1600
1640 DATA 392,300,392,300,440,300,370,450,392,150,440,300
1650 DATA 494,300,494,300,523,300,494,450,440,150,392,300
1660 DATA 440,300,392,300,370,300,392,900,0,0
1670 M$="YOU'RE RICH YOU FOUND THE GOLD"
1680 FOR J=1 TO LEN(M$)
1690 CALL HCHAR(2,1+J,ASC(SEG$(M$,J,1)))
1700 NEXT J
1710 PRINT
1715 CALL HCHAR(TR-1,TC,88)
1720 INPUT "WANT TO PLAY AGAIN? "A$
1730 IF SEG$(A$,1,1)="Y" THEN 280
1740 IF SEG$(A$,1,1)<>"N" THEN 1720
1750 END

```



You could be on our pages

We welcome programs articles and tips from our readers.

PROGRAMS should, if possible, be computer printed to a width of 48 characters (use a new ribbon) and/or sent on cassette. Check carefully that they are bug-free. Include details of what your program does, how it works, variables you have used and hints on conversion.

ARTICLES on using home computers should be no longer than 2,000 words. Don't worry about your writing ability — just try to keep to the style in HCW. Articles most likely to be published will help our readers make better use of their micros by giving useful ideas, possibly with programming examples. We will convert any sketched illustrations into finished artwork.

TIPS are short articles, and brief programming routines which we can put together with others. Your hints can aid other computer users.

Competitive rates are paid.

Keep a copy of your submissions and include an SAE if you want them returned. Label everything clearly and give a daytime and home phone number if you can.

Paul Liptrot, Home Computing Weekly, 145 Charing Cross Road, London WC2H 0EE

Just scrolling Spectrum

Tim Boone mentioned in his tip about scrolling letters, that he did not understand why POKEing 23606 with a number other than 0 scrolled letters.

It works because 23606 and 23607 are the low byte and high byte respectively of the starting address of the character set in the Spectrum (15616) and when converted give values of 0 and 60.

Each character is made up of a set of eight bytes; the first and last of which in each character has the value 0, i.e. a blank line at top and bottom.

When 23606 is POKEd with a value greater than 0, say 1, then the start address of the character set then becomes 15617 instead of 15616. Consequently, all the letters printed on the screen lose the top blank line, which then appears on the letter alphabetically before it, i.e. the blank line at the top of the letter "B" appears at the bottom of the letter "A" moving A up the screen one line and so on through the whole character set.

This also explains why you are able to use a redefined character set in RAM, putting it wherever you need it. The new values for 23607 and 23606 are calculated by:

for 23607: LET HIGH = (INT (ADDRESS/256))
for 23606: LET LOW = (INT (ADDRESS-256*HIGH))

These two values will then indicate the point at which the new character set is in memory.

A. MacLennan

Talking program

As a newcomer to computing (Atari 800), am I alone as a one-fingered typist? When following a program in a magazine with one hand how do you press two keys together, i.e. SHIFT and CLEAR etc? A big hand span? Or lose your place in the program?

If several lines are similar it's easy to go back to the wrong line. I think I may have a solution which may help others.

I use a small portable cassette recorder with a built in mike and read out programs aloud. This has several advantages:

- Two hands to type with (read as two fingers).
- Familiarising oneself with the program layout.

Among these discoveries and suggestions could be just what you were seeking. Why not share your tips? Send them to: Tips, Home Computing Weekly, 145 Charing Cross Road, London WC2H 0EE

- Fewer typing errors.

If the program does not work then it can be a long and tedious job checking through it, but with a quick rewind of the tape you can listen to yourself while checking the screen.

I must admit it took a little while to get my speech down to a speed that I could type at but there is always a pause button or a short rewind.

J. R. Styth

Tuning up the Oric

Here is a routine for playing music using the three channels of the Oric:

TS holds the numbers for the tune.

```
10 TS = "14321432141414321"
20 FOR I = 1 TO LEN(TS)
30 N = ASC(MID$(TS,I,1))-47
40 PLAY 7,0,5,2500
50 MUSIC 1,1,N,10
60 MUSIC 2,2,N,10
70 MUSIC 3,3,N,10
80 WAIT 15
90 NEXT I
100 PLAY 0,0,0,0
```

B. Wilkinson

Defined characters on your 16k VIC

To convert a 16K + VIC to use defined characters, POKE the following.

```
POKE 648,30:POKE 642,32:
SYS 64824 (Return)
```

Enter as a direct command, as this clears the VIC as though you had just switched on and reserves memory below the screen, which is now back at 7680, for defined characters.

But take out any 51,52,53,54,55,56s, as these are no longer needed in a program!

David Shepherdson

Garbage collecting TI-99/4A

One of your reviewers recently mentioned the TI-99/4A pausing while a program was running.

This is the garbage collector.

If you EDIT a program line, there is a pause before the cursor reappears: the computer is removing an old line, moving the others up and adding the new line to the stack.

In a running program it is more efficient to do this only when memory becomes full: in the meantime "dead" variable values are stored. Only when memory is full of redundant values does the computer take time out to throw out the dross. While it does this, everything stops for a short while (even the sprites in Extended Basic).

Garbage collection takes place more frequently the longer the program. There is no way of preventing it, short of avoiding use of the stack (very difficult!).

Stephen Shaw

Keeping TAB Oric

If you fear your Oric may be faulty because the TAB doesn't work, relax — this is perfectly normal. Use SPC in place of TAB and you will find absolutely no difference, as long as you've got a PRINT statement before it.

You can also use EDIT on the Oric, though you won't find it in the user manual anywhere. Perhaps the manual needs editing?

Ian Scott

Private lines Spectrum

Just a simple tip for Spectrum owners who want to make a private program inaccessible to anyone except themselves.

Before SAVEing put in as a direct command:

LET s\$ = "(some secret word, figure, graphic, what have you)"

SAVE from a program line with an auto-run:

7000 SAVE "Private" LINE 5

And use these lines:

5 FOR c = 1 TO 3: CLS: PRINT "Access-word, please?": INPUT

LINE X\$: IF X\$ = s\$ THEN GO TO 15

10 NEXT C: NEW

15 CLS (and go on with the program)

The use of LINE in line 5 makes it impossible to use BREAK or STOP so you cannot LIST, and without the access-word you cannot go on. Try three times and you lose the lot!

After using the program it is as well to change the word by LET s\$ = "(a new word)" and then to SAVE again.

Do remember the word yourself or you've had it!

Andrew Turek

Make a message Oric-1

This subroutine for the 48K Oric-1 allows the user to display messages on the top line of the display (normally reserved for the Oric's messages). This could be useful for displaying scores etc in games, because this line does not scroll with the rest of the page.

```
100 REM ROUTINE TO
PRINT ON TOP LINE
100 M$ = "TEXT TO BE PUT
ON TOP LINE"
120 FOR C = #BB83 TO #
BB83 + 36: POKE C,32:
NEXT C
130 FOR C = 1 TO LEN (M$):
POKE #BB83 + C, ASC
(MID$(M$,C,1)): NEXT C
140 RETURN
```

The text can be made flash by adding:

```
125 M$ = CHR$(#C) + M$
```

If any reader has a 16K Oric then 8000 HEX must be subtracted from all addresses used.

Colin Failes

Testing, testing TI-99/4A

Most of the assembly language games on TI-99/4A modules have a test mode. Return to the game title screen using BACK, then quickly press keys *#* (eg hold SHIFT down and press 8 3 8). Then respond to the screen prompts quickly.

Stephen Shaw

Think carefully — you may lose your computer

Micro Checkers is a game of strategy for two players. The object is to capture and destroy your opponent's microcomputer.

You have seven chips to do battle with. These must not only be used for attacking your enemy's micro, but also for defending your own. Players may move any of their pieces (including their micro), one square diagonally in any direction, on each of their turns.

You select which player to move, and where to move it to, by pressing the appropriate letter keys as prompted on the screen.

To capture an opponent's piece, including their micro, you simply move to the square that

Hints on conversion

CALL CHAR(ASCII code number, hexadecimal string) defines the pattern to be given to the character with the ASCII code stated

CALL HCHAR(row number, column number, ASCII code, number of repetitions) places character with ASCII stated, at row and column specified, and optionally repeats it horizontally the number of times stated

CALL VCHAR(row number, column number, ASCII code, number of repetitions) works in the same way as CALL HCHAR only repeats vertically

CALL GCHAR(row number, column number, numeric variable) equivalent of PEEK. Assigns ASCII code of character at row and column stated to numeric variable

CALL CLEAR clears the screen

CALL COLOR(character set, foreground colour code, background colour code) specifies foreground and background colours for all characters in set specified

CALL SCREEN(colour code) specifies screen colour using colour code

CALL SOUND(duration, frequency, volume) produces sound. Duration in milliseconds, frequency in Hertz (or noise code), and volume in scale from 0 loudest to 30 quietest

CALL KEY(key unit,k,s) equivalent of INKEYS, returns code of key pressed in variable k

Micros are the targets in this two-player strategy game written by Lance Booth for the standard TI-99/4A

```

100 REM MICRO CHECKERS
110 REM BY LANCE BOOTH
120 CALL CHAR(35,"*****")
130 RESTORE 100
140 FOR I=144 TO 151
150 READ C#
160 CALL CHAR(I,C#)
170 NEXT I
180 DATA 001F101010101F,00F000000000F0,03031F334C737F,C0C0F034CE32FE
190 DATA 00000000F03030F,000000C0F0C0C0F0,03030F03,C0C0F0C0
200 RESTORE 100
210 FOR I=152 TO 159
220 READ C#
230 CALL CHAR(I,C#)
240 NEXT I
250 CALL COLOR(15,5,1)
260 CALL COLOR(16,10,1)
270 CALL SCREEN(16)
280 CALL CLEAR
290 FOR R=4 TO 16 STEP 4
300 FOR C=8 TO 20 STEP 4
310 CALL HCHAR(R,C,35,2)
320 CALL HCHAR(R+1,C,35,2)
330 CALL HCHAR(R+2,C+2,35,2)
340 CALL HCHAR(R+3,C+2,35,2)
350 NEXT C
360 NEXT R
370 X=10
380 Y=22
390 FOR R=4 TO 6 STEP 2
400 FOR C=X TO Y STEP 4
410 CALL HCHAR(R,C,140)
420 CALL HCHAR(R,C+1,140)
430 CALL HCHAR(R+1,C,150)
440 CALL HCHAR(R+1,C+1,151)
450 CALL HCHAR(R+2,C,156)
460 CALL HCHAR(R+2,C+1,157)
470 CALL HCHAR(R+3,C,158)
480 CALL HCHAR(R+3,C+1,159)
490 NEXT C
500 X=X+2
510 Y=Y+20
520 NEXT R
530 CALL HCHAR(4,14,144)
540 CALL HCHAR(4,15,145)
550 CALL HCHAR(5,14,146)
560 CALL HCHAR(5,15,147)
570 CALL HCHAR(18,16,152)
580 CALL HCHAR(18,17,153)
590 CALL HCHAR(19,16,154)
600 CALL HCHAR(19,17,155)
610 FOR R=3 TO 20 STEP 17
620 C#
630 FOR CH=72 TO 86 STEP 2
640 CALL HCHAR(R,C,CH)
650 C=C+2
660 NEXT CH
670 NEXT R
680 FOR C=7 TO 24 STEP 17
690 R=5
700 FOR CH=69 TO 83 STEP 2
710 CALL HCHAR(R,C,CH)
720 R=R+2
730 NEXT CH
740 NEXT C
750 REM GAME SEQUENCE
760 CH=150
770 GOSUB 810
780 CH=158
790 GOSUB 810
800 GOTO 750
810 REM MOVE ROUTINE
820 M#="WHERE FROM ROW FIRST"
830 GOSUB 1050
840 GOSUB 1060
850 IF (GET=35)+(GET=32) THEN 1210
860 IF (GET<>CH)+(GET<>CH-4) THEN 1250
870 REM
880 REM
890 CALL HCHAR(R-1,C,32,2)
900 CALL HCHAR(R,C,32,2)
910 OR=R
920 OC=C
930 GET=GET
940 M#="WHERE TO ROW FIRST"
950 GOSUB 1050
960 GOSUB 1060
970 IF (R<OR-2)+(R>OR+2)+(C<OC-2)+(C>OC+2) THEN 1210
980 IF GET=CH-4 THEN 1290
990 IF (GET=CH+4)+(GET=CH-12) THEN 1300
1000 IF GET=35 THEN 1210
1010 CALL HCHAR(R-1,C,GET-2)
1020 CALL HCHAR(R-1,C+1,GET-1)
1030 CALL HCHAR(R,C,GET)
1040 CALL HCHAR(R,C+1,GET+1)
1050 RETURN
1060 REM ROW/COL CHOICE
1070 CALL KEY(4,R,SR)

```

How it works

120-260 define characters and colours
270 sets screen colour
280 clears screen
290-360 displays board
370-600 displays players
610-740 displays square guide
750-800 game loop
810-1050 move routine
820-840 select old square
850-880 situation check
890-900 clear old position
910-930 swap variables
940-960 select new square
970-1000 situation check
1010-1040 display new player
1060-1190 select square
1070-1100 select row
1110-1120 delay loop
1130-1160 select column
1170 is square on board
1180 look at new position
1200-1540 what happened
1210-1240 bad square
1250-1280 bad player
1290-1320 destroyed own micro
1330-1370 plays music
1380-1540 winner
1550-1670 displays messages
1680-1760 play again?

piece is occupying. Be careful where you move — you might destroy one of your own pieces or leave yourself open to attack.

If you make an illegal move or don't follow instructions or try to move on of your opponent's pieces than you will be disqualified and will lose the game.

This program was written for the standard TI-99/4A but should be convertible for use on any computer with user-defined characters.

```

1660 NEXT J
1670 RETURN
1680 REM PLAY AGAIN
1690 M#="WANT TO PLAY AGAIN?(Y OR N)"
1700 FOR J=1 TO LEN(M#)
1710 CALL HCHAR(23,1+J,ASC(SEG(M#,J,1)))
1720 NEXT J
1730 CALL KEY(3,K,S)
1740 IF S=0 THEN 1730
1750 IF K=89 THEN 280
1760 IF K>78 THEN 1730
1770 END

```



Kitten Kong, a 10-foot-high mutant tabby, is playing cat-and-mouse with a distressed damsel.

You must go to the rescue by running along girders and up ladders, jumping over the gaps. If you miss your footing or run off the end of a girder, you plunge to your death.

You move left and right using the S and D keys. Use E to go up a ladder or jump, depending on whether a ladder is above you or not. And use X to go down a ladder or stay still, depending on whether there's a ladder.

You get 10 points for reaching the damsel, and may gain extra points if you jump and reach the 1s.

From time to time Kong leaps into the air to scare you, and if you are on one of the top two levels of girder then you will fall off.

If you manage to rescue the damsel this time then there's another level waiting for you — with one piece of girder missing.

How it works

100-270 sets colour and characters
280-540 sets screen
550 sets variables
570-620 Kong jumps
650-670 damsel moves
680-790 you move
820-900 you jump
910-950 up ladder
960-1010 down ladder
1020-1070 you fall
1080-1160 sets up next level
1170-1180 you lose
1190-1230 play again or end

Fend off the ferocious feline

When a cuddly kitten turns into a marauding monster, you can't just stand there and let a damsel in distress be turned into cat food. Paul Edwards explains the rescue plan for his strangely familiar game, written for the TI-99/4A plus Extended BASIC



Hints on conversion

Here are explanations of some commands in TI BASIC, all prefaced by the word CALL.

SPRITE sets up a sprite of number x, ASCII code y, colour z, row r, column c, row velocity rv and column velocity cv

PATTERN makes sprite number x ASCII code y

DELSprite deletes sprite number x

MOTION sets sprite number x in motion with row velocity rv and column velocity cv

Variables

R player's row
C player's column
PR player's previous row
PC player's previous column
MI, MZ two characters for walking man
DC damsel in distress's column
DD damsel in distress's direction
CH character of position player is about to move on to
CCH the last character covered — printed offer player has passed
SC score
K ASCII code from CALL KEY
D,E characters from CALL GCHAR

```
100 CALL CLEAR :: CALL SCREEN(2)
110 FOR Q=0 TO 8 :: CALL COLOR(Q,16,1):: NEXT Q
120 RANDOMIZE
130 CALL CHAR(96,"181808181C180818")
140 CALL CHAR(97,"181808181C185422")
150 CALL CHAR(98,"1818101838181018")
160 CALL CHAR(99,"1818101838182A44")
170 CALL CHAR(100,"1C1C083E081C1436")
180 CALL CHAR(101,"000000000000BFF")
190 CALL CHAR(104,"243C243C243C243C")
200 CALL CHAR(124,"AA55AA55")
210 CALL CHAR(128,"01030303070F1F3B")
220 CALL CHAR(129,"80C0C0C0E0F0F8DC")
```

```
230 CALL CHAR(130,"373F1F070E0E060E")
240 CALL CHAR(131,"ECFCF8F07070607")
250 CALL CHAR(136,"1C1C087F1C1C3E14")
260 CALL CHAR(137,"08180808")
270 CALL COLOR(9,12,1,10,13,1,12,5,1,13,14,1,14,8,1)
280 CALL HCHAR(22,1,124,32)
290 DISPLAY AT(4,13):"IIII"
300 DISPLAY AT(7,11):"IIIIIIII"
310 DISPLAY AT(10,7):"I III III IIIII"
320 DISPLAY AT(13,4):"III I II I III I IIII"
330 DISPLAY AT(16,1):"IIIIII III I III I III I IIII"
340 DISPLAY AT(19,4):"IIIIII II II I IIIII"
350 CALL VCHAR(4,14,104,3)
360 CALL VCHAR(4,19,104,3)
370 CALL VCHAR(7,12,104,3)
380 CALL VCHAR(7,21,104,3)
390 CALL VCHAR(10,8,104,3)
400 CALL VCHAR(10,14,104,3)
410 CALL VCHAR(10,19,104,3)
420 CALL VCHAR(10,26,104,3)
430 CALL VCHAR(13,5,104,3)
440 CALL VCHAR(13,12,104,3)
```


TI-99/4A PROGRAM

```

450 CALL VCHAR(13,17,104,3)
460 CALL VCHAR(13,29,104,3)
470 CALL VCHAR(16,9,104,3)
480 CALL VCHAR(16,21,104,3)
490 CALL VCHAR(19,15,104,3)
500 CALL VCHAR(19,26,104,3)
510 CALL HCHAR(5,16,128):: CALL
HCHAR(5,17,129):: CALL HCHAR(6,1
6,130):: CALL HCHAR(6,17,131)
520 CALL HCHAR(8,14,137):: CALL
HCHAR(8,20,137):: CALL HCHAR(11,
13,137):: CALL HCHAR(11,22,137)
530 CALL HCHAR(14,8,137):: CALL
HCHAR(14,25,137):: CALL HCHAR(17
,12,137):: CALL HCHAR(17,18,137)
540 CALL HCHAR(20,10,137):: CALL
HCHAR(20,23,137)
550 PR,R=21 :: PC,C=1 :: M1=98 :
: M2=99 :: DC=15 :: DD=1 :: CCH=
0
560 IF RND>.05 THEN 640
570 CALL HCHAR(5,16,32,2):: CALL
HCHAR(6,16,32,2)
580 CALL HCHAR(4,16,128):: CALL
HCHAR(4,17,129):: CALL HCHAR(5,1
6,130):: CALL HCHAR(5,17,131)
590 CALL HCHAR(4,16,32,2):: CALL
HCHAR(5,16,32,2)
600 CALL HCHAR(5,16,128):: CALL
HCHAR(5,17,129):: CALL HCHAR(6,1
6,130):: CALL HCHAR(6,17,131)
610 CALL SOUND(500,-5,0,110,0)
620 CALL HCHAR(4,16,124,2)
630 IF R=6 OR R=3 THEN 1020
640 CALL HCHAR(R,C,M1):: DISPLAY
AT(1,1):"SCORE : -";SC
650 CALL HCHAR(3,DC,32)
660 DC=DC+DD :: IF DC=15 OR DC=1
8 THEN DD=-DD
670 CALL HCHAR(3,DC,136)
680 PR=R :: PC=C
690 CALL KEY(0,K,S)
700 IF K=83 THEN C=C-1 :: M1=96
:: M2=97
710 IF K=68 THEN C=C+1 :: M1=98
:: M2=99
720 IF PR=R AND PC=C AND K<>69 A
ND K<>88 THEN IF M1=96 THEN C=C-
1 ELSE IF M1=98 THEN C=C+1
730 IF C=0 THEN C=32 ELSE IF C=3
3 THEN C=1
740 CALL GCHAR(R,C,CH)
750 IF CH=136 THEN 1080 ELSE IF
CH>127 AND CH<132 THEN C=PC :: G
OTO 1170
760 CALL HCHAR(PR,PC,CCH)

```

```

770 CCH=CH :: CALL HCHAR(R,C,M2)
780 CALL GCHAR(R+1,C,D):: IF D=3
2 THEN 1020
790 IF K=69 THEN 800 ELSE IF K=8
8 THEN 960 ELSE 560
800 CALL GCHAR(R-1,C,D):: IF D=1
04 THEN 910
810 IF C<3 OR C>29 THEN CCH=32 :
: GOTO 560
820 CALL GCHAR(R-1,C-1,D)
830 IF D=137 THEN SC=SC+1 :: CAL
L SOUND(100,800,0):: D=32
840 IF M1=96 THEN CALL GCHAR(R,C
-2,E):: CALL HCHAR(R,C,32):: CAL
L HCHAR(R-1,C-1,M2):: CALL HCHAR
(R-1,C-1,D):: CALL HCHAR(R,C-2,M
1):: C=C-2
850 CALL GCHAR(R-1,C+1,D)
860 IF D=137 THEN SC=SC+1 :: CAL

```



```

L SOUND(100,800,0):: D=32
870 IF M1=98 THEN CALL GCHAR(R,C
+2,E):: CALL HCHAR(R,C,32):: CAL
L HCHAR(R-1,C+1,M2):: CALL HCHAR
(R-1,C+1,D):: CALL HCHAR(R,C+2,M
2):: C=C+2
880 CCH=E
890 CALL GCHAR(R+1,C,D):: IF D=3
2 THEN 1020
900 GOTO 560
910 FOR RR=R TO R-2 STEP -1
920 CALL HCHAR(RR,C,104):: CALL
HCHAR(RR-1,C,100)
930 NEXT RR
940 R=R-3 :: CCH=32
950 GOTO 560
960 CALL GCHAR(R+1,C,D)
970 IF D<>104 THEN CCH=32 :: GOT

```


TI-99/4A PROGRAM

```

0 560
980 CALL HCHAR(R,C,32)
990 FOR RR=R+1 TO R+3 :: CALL HC
HAR(RR,C,100):: CALL HCHAR(RR,C,
104):: NEXT RR
1000 R=R+3 :: CCH=104
1010 GOTO 560
1020 CALL HCHAR(R,C,32)
1030 CALL SPRITE(#1,M1,12,R*8,C*
8-8,10,0)
1040 CALL POSITION(#1,Y,X):: IF
Y<160 THEN 1040
1050 CALL MOTION(#1,0,0):: CALL
PATTERN(#1,101)
1060 FOR Q=400 TO 200 STEP -10 :
: CALL SOUND(-100,Q,0):: NEXT Q
1070 GOTO 1190
1080 FOR Q=200 TO 400 STEP 10 ::
CALL SOUND(-100,Q,0):: NEXT Q
1090 SC=SC+10
1100 Y=INT(RND*13)+9 :: X=INT(RN

```

```

D*32)+1 :: CALL GCHAR(Y,X,D):: I
F D<124 THEN 1100
1110 CALL HCHAR(Y,X,32)
1120 FOR Q=1 TO 500 :: NEXT Q
1130 CALL HCHAR(R,PC,32)
1140 CCH=32
1150 CALL DELSPRITE(#1)
1160 GOTO 520
1170 FOR Q=400 TO 200 STEP -10 :
: CALL SOUND(-100,Q,0):: NEXT Q
1180 DISPLAY AT(2,1):"HARD LUCK
YOU LOSE !!!"
1190 DISPLAY AT(24,1):"PLAY AGAI
N ? (Y/N)"
1200 CALL KEY(0,K,S)
1210 IF K=89 THEN SC=0 :: CALL D
ELSPRITE(#1):: CALL CLEAR :: GOT
0 280 ELSE IF K=78 THEN 1220 ELS
E 1200
1220 CALL CLEAR :: END

```

TI 99/4A
Home Computer
only £99.50

TEXAS INSTRUMENTS

TI 99/4A Solid State Software

Some of our Crazy Prices:

EXTENDED BASIC £52.25
MINI MEMORY £52.25

BLASTO £10.50
CHISHOLM TRAIL £16.80
TUNNELS OF DOOM £20.80
SOCCER £12.50

Post & Packing Free - VAT included
Send S.A.E. for full price list
Please make cheques payable to

PARCO ELECTRICS

4 Dorset Place, New Street, Honiton,
Devon, EX14 8GS.
Tel: Honiton (0404) 44425

Dragon 32 — Spectrum — Tandy Color

FREE TAPE OFFER

CODE		
D/T	Lost in Space	
D/T	Draculas Castle	
D/T	Floating Bomb	ALL £5
D/T	Death Ship	

BUY ONE GET ONE FREE

D/T	A-MAZE-IN	£10
	3D Maze with death defying monsters — Hours of fun for all — different game each time.	
	Buy a copy of A-Maze-In and get any TWO of the above mentioned FREE	
	"Good Value for Money" S. Drew, York	
D/T/S(48)	Ownership	£7
	Very realistic version of Monopoly with all the thrills and more over 24K	
S(16)	Games pack No. 1 (over 30K of programs)	£4
	Bomb Run, Yellow Oil, Monster and more	
	Games pack No. 2 (over 30K of programs)	£4
	Snake, Zombie Lander and more	
	"Liked games pack one send me No. 2. C. W. Thom, Glasgow"	

Buy both for £6.50

BIG CASH PRIZES

D/T/S(16)	Roving Knight	£5
	(includes entry to competition).	
	This amazing board challenge has still to be cracked.	
	Try your luck and you could be pounds in	

Code D = Dragon 32 T = Tandy S = Spectrum 16K or 48K

STATE WHICH MACHINE

Send cheques or Postal orders to:

FLATBELL Ltd

9 Franklin Road

Haddenham, Bucks HP17 8LE



Adrian Hedley and Jo Wheeler will present Central TV's Magic Micro Mission

A mission to demystify the micro

Central TV is jumping on the home computer bandwagon with The Magic Micro Mission, a new TV series that will emphasise home computer games.

Magic Micro Mission will, Central says, explore what's happening in computer technology today as well as looking ahead to future developments, and is pledged to demystify the computer world — for example, by explaining computer terms like ROM and RAM.

Each week, a panel of five children will try out a range of hardware and software.

Presenting the show are Jo Wheeler and Adrian Hedley, plus resident expert John Barker of Warwick University, and they

will be joined each week by a number of well-known names.

The first programme will feature Dave Lee Travis and Patrick Bossert of Rubik's Cube fame, plus Willy Rushton and Chris Sievey. Peter Choy of The Chinese Detective, whose hobby is making Lego robots controlled by his Spectrum, will make a guest appearance on the second programme. And Rick Wakeman will appear on the third.

If you live in the East of West Midlands, you'll be able to tune into the half-hour programme at 5.15 every Wednesday from 9 November. The series will run for six weeks, and may be repeated before Christmas. Central is hoping that other TV regions will screen it too.

Central TV, Central House, Broad Street, Birmingham B1 2JP

New club for TI owners

TI owners now have a national users group, which takes over from the old TIhome.

Called the TI Home Computer Users Club, it aims to offer the same benefits as TIhome, but at a reduced fee of £5 per year instead of £12.

This reduction in fees will be possible because the new club, run by Ray Hodges of Hodges Associates, is aiming at mass membership. TI has given the club access to its own mailing list, and will be putting club information in with all new TI-99/4As sold.

In addition, TI will be supporting the club with new product information and technical advice, "though not financially", according to TI's Robin Frowd. "It's not going to be a mouthpiece for TI".

For their £5, members receive a membership card, a quarterly magazine and details of local TI user groups.

They will also have access to the TIhome software library, which offers a mail order service for software from all over the world, as well as books and listings.

Paul Dicks, who set up TIhome, will continue to be involved with the new club, running the software library and contributing to the magazine.

Ray Hodges, the new organiser, is no stranger to organising clubs — she set up the first Atari VCS users group in the UK. She will have full-time staff who can answer membership queries on the spot, and get answers to technical queries. Members can ring the club on 0628 71696.

TI Home Computer Users Club, PO Box 190, Maidenhead, Berks SL6 1YX

Why programs need a seal of approval

Microcomputers, we are constantly being told, will bring about a revolution in the classroom.

A prerequisite of this revolution will be the ready availability of top quality software, fully documented and error trapped and free from bugs.

Anyone who takes the trouble to read the software reviews in this and other magazines will know that there is not, at present, a great deal of educational software available, and a good proportion of what there is at best of dubious educational benefit.

In the short time I have been involved in software retailing, it has become transparently clear to me that parents of children who use micros at school or at home are crying out for some decent software to lure their children away from Space Invaders and the like.

At the moment there is no way for them to check the quality of the software that they are thinking of buying (unless they cut out all the reviews and file them away for future reference).

It would therefore seem like a very good idea to have some sort of universal standard by which educational software can be measured.

Having given the subject a great deal of thought I have come up with the following ideas:

- 1 That representatives of the microcomputing industry (particularly from the software side) approach the Department of education with a view to setting up a joint panel of teachers and programmers to identify the areas in which microcomputers can be most beneficial.
- 2 Having identified those areas, produce a set of guidelines for aspiring software authors in order to achieve some uniformity of presentation.
- 3 Anyone wanting their software to carry Department of Education approval would have to submit their programs to the panel and carry out any changes that the panel decide are necessary. Then, and only then, would authors be allowed to advertise their programs as "approved educational software."

It is my belief that the availability of software carrying such approval would encourage parents who are contemplating buying their children a micro to go ahead with the purchase, secure in the knowledge that it will help them to learn as well as giving them many hours of entertainment.

Obviously a great deal of discussion would need to take place before my ideas could be put in to practice but it is essential that some sort of debate should start now if our children are to reap the rewards that the microcomputer offers.

Graham Barrow
Partner, Sheensoft Software,
Richmond,
Surrey

You're alone . . . and here come the invaders

Waves of invaders are on their way to conquer the planet. Only you, armed with a laser cannon, can fight them off.

My Texas Invaders game runs on the unexpanded TI-99/4.

To ensure the game is fast moving, the invaders do not move and sound is kept to a minimum.

But if you do happen to miss an invader you lose one of your three lives.

How it works

10-20 clear screen and turn it black
30-360, 1180-1260, 1470,1540 define characters, assign colours
370-850 title screen
860-990 ask if instructions needed
1000-1170 print instructions
1270-1290 set up variables
1300-1460 set up first screen
1550 set up variable for number of invaders
1560-1620 set up screen
1630-1690 detect keys, branch to movement, fire routines
1700-1840 left, right base movement routines
1850-1920 fire, detect hit routine
1930-2060 hit routine, branch to miss routine if no hits
2070-2120 if miss deduct life
2130-2180 add 10 to score
2190-2350 if all lives lost ask if new game wanted

**Earth is under threat again.
Armed with a laser cannon, you
face hordes of enemy fighters.
Type in Eric Liddell's game for
the TI-99/4A before we're
conquered**

```
10 CALL CLEAR
20 CALL SCREEN(2)
30 RESTORE 80
40 FOR I=48 TO 58
50 READ A$
60 CALL CHAR(I,A$)
70 NEXT I
80 DATA 007E7E66667E7E,003838181
83C3C,007E7E0E707E7E,007E063E067
E7E,0060606C7E7E0C,007E7E700E7E7
E,007E607E667E7E
90 DATA 007E7E66060606,003C243C6
67E7E,007E7E667E7E06,3C5A187E7E
74224
100 RESTORE 150
110 FOR I=65 TO 73
```

```
120 READ A$
130 CALL CHAR(I,A$)
140 NEXT I
150 DATA 003C24247E6666,183C7EDB
DBC366,007E424060627E,007E426262
627E,007E40407C607E,007E40407C60
6
160 DATA 007E424066627E,00242424
7E6666,00101010181818
170 CALL CHAR(75,"004444786464")
180 CALL CHAR(76,"0040406060607E
")
190 RESTORE 240
200 FOR I=78 TO 89
210 READ A$
220 CALL CHAR(I,A$)
230 NEXT I
240 DATA 007E4262626262,007E4246
46467E,007E42427E606,007E4246464
67F01,007C44447E6262,007E42701E4
67E
250 DATA 007E1010181818,00424262
62627E,00626262222418,0062626A6A
4A76,00426618186642,004242241818
```

```
18
260 CALL COLOR(3,6,2)
270 CALL COLOR(4,6,2)
280 FOR I=5 TO 8
290 CALL COLOR(I,5,2)
300 NEXT I
310 CALL CHAR(96,"08022E9FBFFFF3
7F")
320 CALL CHAR(97,"104074F9FDFFCF
FE")
330 CALL CHAR(98,"3F150F07224224
18")
340 CALL CHAR(99,"FC50F870222112
0C")
350 CALL CHAR(100,"183C7EE73C5AC
381")
360 CALL COLOR(9,9,2)
370 CALL HCHAR(4,15,96)
380 CALL SOUND(-10,110,0)
390 CALL HCHAR(4,16,97)
400 CALL SOUND(-10,115,0)
410 CALL HCHAR(5,15,98)
420 CALL SOUND(-10,110,0)
430 CALL HCHAR(5,16,99)
```

Variables

MISS number of times shot has failed to hit invader
Sc score
c column position of laser base
NV number of invaders. When NV equals zero a new wave appears

Hints on conversion

TI BASIC is fairly standard apart from its graphics. Characters are defined on an eight grid using the CALL CHAR command followed by a hexadecimal string. Colours are set by using the CALL COLOR command.

Here are some of the commands I have used in the program:

CALL HCHAR and CALL VCHAR position a character at specified row and column on the screen and optionally repeat it horizontally or vertically

CALL GCHAR tests specified character positions on the screen to see if they are occupied

CALL SCREEN sets the screen to any of 16 colours

CALL CLEAR is a direct equivalent of CLS (clear screen)

TI-99/4A PROGRAM

```

440 CALL SOUND(100,120,5)
450 T$="          TEXAS"
460 R=7
470 GOSUB 520
480 T$="          INVADERS"
490 R=9
500 GOSUB 520
510 GOTO 570
520 FOR I=1 TO LEN(T$)
530 CALL HCHAR(R,I,ASC(SEG$(T$,I,1)))
540 CALL SOUND(-10,3000,0)
550 NEXT I
560 RETURN
570 FOR I=1 TO 32 STEP 2
580 FOR R=15 TO 20 STEP 5
590 CALL HCHAR(R,I,100)
600 CALL SOUND(-100,115,10)
610 CALL HCHAR(R,I,66)
620 CALL SOUND(-100,110,15)
630 NEXT R
640 NEXT I
650 FOR I=1 TO 100
660 NEXT I
670 FOR I=57 TO 48 STEP -1
675 CALL SOUND(-10,4300,0)
676 CALL SOUND(-10,5000,0)
680 CALL HCHAR(12,16,I)
690 FOR D=1 TO 200
700 NEXT D
720 NEXT I
725 CALL HCHAR(12,16,100)
730 FOR I=0 TO 30 STEP 2
740 CALL SOUND(-10,110,15,110,15,220,15)
750 CALL SOUND(-100,110,I,110,I,220,I)
760 CALL SOUND(-100,110,I,220,I,330,I)
770 CALL SOUND(-100,220,I,220,I,110,I)
780 CALL SOUND(-100,110,I,110,I,220,I)
790 NEXT I
800 P$="          PRESS ANY KEY TO CONTINUE"
810 FOR I=1 TO LEN(P$)
820 CALL HCHAR(22,I,ASC(SEG$(P$,I,1)))
830 NEXT I
840 CALL KEY(0,K,S)
850 IF S=0 THEN 840
860 CALL CLEAR
870 I$="          DO YOU WANT INSTRUCTIONS"
880 Y$="          Y OR N"

```

```

890 FOR I=1 TO LEN(I$)
900 CALL HCHAR(10,I,ASC(SEG$(I$,I,1)))
910 NEXT I
920 FOR I=1 TO LEN(Y$)
930 CALL VCHAR(I,16,ASC(SEG$(Y$,I,1)))
940 NEXT I
950 CALL KEY(0,K,S)
960 IF S=0 THEN 950
970 CALL CLEAR
980 IF K=ASC("Y") THEN 1000
990 IF K=ASC("N") THEN 1180 ELSE 950
1000 CALL CLEAR
1010 I$="          INSTRUCTIONS"
1020 R=5
1030 GOSUB 1110
1040 I$="          KEYS      S LEFT      D R
IGHT"
1050 R=10
1060 GOSUB 1110
1070 I$="          USE FULL STOP TO FIRE"
1080 R=13
1090 GOSUB 1110
1100 GOTO 1180
1110 FOR I=1 TO LEN(I$)
1120 CALL HCHAR(R,I,ASC(SEG$(I$,I,1)))
1130 NEXT I
1140 FOR I=1 TO 200
1150 NEXT I
1160 CALL CLEAR
1170 RETURN
1180 RESTORE 1230
1190 FOR I=104 TO 115
1200 READ A$
1210 CALL CHAR(I,A$)
1220 NEXT I
1230 DATA FFFFFFFFFFFFFFFFFF,7F3F392C8727491,FEFC9C32E1E29408,FF82F4081F3C7FFC,FF412F10F83CFE3F,F8F0F8F8FCFDFFFF
1240 DATA 0085FFFFFFFFFFFFFF,24242424242DBFFF,0F1F0F0787CFFFFFF,0080C0E0E0F0F0F8,010303010103070F,F1824429924C324
1250 CALL COLOR(10,13,2)
1260 CALL COLOR(11,13,2)
1270 MISS=4
1280 SC=-10
1290 C=16
1300 GOSUB 2070
1310 GOSUB 2130
1320 CALL HCHAR(24,1,104,32)

```


TI-99/4A PROGRAM

```

1330 CALL HCHAR(23,1,104)
1340 CALL HCHAR(23,32,104)
1350 CALL HCHAR(23,2,109)
1360 CALL HCHAR(23,31,112)
1370 CALL HCHAR(23,4,111,26)
1380 CALL HCHAR(22,4,115,26)
1390 CALL HCHAR(22,1,104)
1400 CALL HCHAR(22,2,113)
1410 CALL HCHAR(21,1,113)
1420 CALL HCHAR(22,31,114)
1430 CALL HCHAR(22,32,104)
1440 CALL HCHAR(21,32,114)
1450 CALL HCHAR(22,3,107)
1460 CALL HCHAR(22,30,108)
1470 CALL CHAR(144,"000018243C7E
FF3C")
1480 CALL CHAR(145,"423C66FF7E24
42")
1490 CALL COLOR(15,11,2)
1500 CALL CHAR(152,"101008081010
0808")
1510 CALL CHAR(153,"001004104804
1")
1520 CALL CHAR(154,"482448946214
2892")
1530 CALL CHAR(155,"8822954834AA
449")
1540 CALL COLOR(16,16,2)
1550 NU=44
1560 FOR I=7 TO 27 STEP 2
1570 CALL HCHAR(6,I,145)
1580 CALL HCHAR(9,I,100)
1590 CALL HCHAR(12,I,58)
1600 CALL HCHAR(15,I,66)
1610 NEXT I
1620 CALL HCHAR(21,C,144)
1630 CALL KEY(0,K,S)
1640 IF S=0 THEN 1630
1650 IF MISS=0 THEN 2190
1660 IF NU=0 THEN 1550
1670 IF K=83 THEN 1700
1680 IF K=68 THEN 1780
1690 IF K=46 THEN 1850
1700 C=C-1
1710 CALL HCHAR(21,C,144)
1720 CALL HCHAR(21,C+1,32)
1730 IF C<4 THEN 1760
1740 GOTO 1630
1750 GOTO 1630
1760 C=4
1770 GOTO 1630
1780 C=C+1
1790 CALL HCHAR(21,C,144)
1800 CALL HCHAR(21,C-1,32)
1810 IF C>29 THEN 1830
1820 GOTO 1630
1830 C=29

```

```

1840 GOTO 1630
1850 FOR I=20 TO 4 STEP -1
1860 CALL GCHAR(I,C,SS)
1870 IF SS<>32 THEN 1930
1880 CALL HCHAR(I,C,152)
1890 CALL HCHAR(I,C,32)
1900 IF I=4 THEN 2050
1910 NEXT I
1920 GOTO 1630
1930 CALL HCHAR(I,C,32)
1940 FOR G=153 TO 155
1950 CALL HCHAR(I,C,G)
1960 NEXT G
1970 FOR G=155 TO 153 STEP -1
1980 CALL SOUND(-200,-7,0)
1990 CALL HCHAR(I,C,G)
2000 NEXT G
2010 CALL HCHAR(I,C,32)
2020 NU=NU-1
2030 GOSUB 2130
2040 GOTO 1630
2050 GOSUB 2070
2060 GOTO 1630
2070 MISS=MISS-1
2080 L$=" LIVES "&STR$(MISS)
2090 FOR I=1 TO LEN(L$)
2100 CALL HCHAR(1,I,ASC(SEG$(L$,
I,1)))
2110 NEXT I
2120 RETURN
2130 SC=SC+10
2140 S$=" SCORE "&STR$(SC)
2150 FOR K=1 TO LEN(S$)
2160 CALL HCHAR(3,K,ASC(SEG$(S$,
K,1)))
2170 NEXT K
2180 RETURN
2190 CALL HCHAR(21,C,32)
2200 FOR I=0 TO 30
2210 CALL SOUND(-200,110,I,-7,I)
2220 NEXT I
2230 CALL SOUND(-10,3000,0)
2240 A$=" ANOTHER GAME Y
OR N"
2250 FOR I=1 TO LEN(A$)
2260 CALL HCHAR(12,I,ASC(SEG$(A$,
I,1)))
2270 NEXT I
2280 CALL KEY(0,K,S)
2281 CALL HCHAR(2,1,32,32)
2290 IF S=0 THEN 2280
2300 IF K=ASC("Y") THEN 2320
2310 IF K=ASC("N") THEN 2340 ELSE
2280
2320 CALL HCHAR(12,1,32,32)
2330 GOTO 1270
2340 CALL CLEAR
2350 END

```


LETTERS

Graphics up and running

You recently carried a newsitem (HCW 29) mentioning that our PowerGraphics program was incompatible with the new Series 3 Spectrums.

We'd just like to say that all our software is now available for any type of Spectrum. All customers who returned un-runable tapes to us have now had them replaced.

Owners of Series 3 Spectrum might like to note that the IN function will malfunction if there is a signal at the cassette port — and also, in certain circumstances, after a BORDER command.

R. L. Butterfield, Buttercraft Software, Keighley, Yorks

Out of its element

Thank you for reviewing one of our Chemistry programs. Element Serpent.

Despite scoring a superb 90 per cent for the instructions, your reviewer failed to understand the object of the game!

The program is used to reinforce the formula concept.

The object of the game is to direct an element eating snake around the screen digesting elements until all the elements present in a compound named at the bottom of the screen are eaten.

Should you eat the wrong element symbol, collide with the wall, or kill the snake the game ends. Wrong symbols are corrected before the game begins again.

Simply really!!!

I hope that this letter clarifies any confusion.

H. K. Farmer, Cloud 9 Software, Tavistock, Devon

Too harsh on Panic

I read with interest your review for the Panic 64 game in issue 30 of Home Computing Weekly, which dropped through my door about an hour after we had just purchased the game for our Commodore 64.

We also experienced the same problem of the sprite corrupting after about an hour's play, but contrary to your report we have found it to be very entertaining.

I think you were unduly harsh when reviewing the game which, when all is said and done, is only an arcade game and meant to be purely entertainment.

With regard to the circuit

Send your letter to Letters, Home Computing Weekly, 145 Charing Cross Road, London WC2H 0EE. Don't forget to name your computer — the best letter could win £5-worth of software. Queries cannot be answered on this page



your monsters managed to get themselves into, we also realised this but the answer is to get back down to the level on which they are gyrating and entice them up to the level you need, just giving your man enough time to get out of the way before they fall through the holes.

Another method is to lead them through the holes and just run away on landing, whereupon they will splat behind you.

Our highest score in just under a week of ownership is 2,200, but I'm not sure how the system works after you go on to level five as there are only four levels shown on the screen.

The highest we have been is level seven.

Sheila Wright, Peterborough

Beyond the Invaders

As an Atari 400 owner I am pleased to see the new Atari computers — the 600XL (16K RAM, £160) and the 800XL (64K RAM, £250).

Both come with typewriter-style keyboards and one cartridge slot. Unlike most new computers they already have an excellent software backup because they are compatible with almost all existing Atari software.

Each has a parallel interface port and external processor bus, the latter for future peripherals including a Z80 CP/M module.

In addition, an expander box will be available, providing three interfaces and slots for eight expansion boards. These include IBM and Z80 interfaces, Winchester hard disc controller and a voice recognition system.

And for those of you that disapprove of Atari BASIC, a faster and better structured disc-based one is available — BASIC A+.

Can anyone now call this a games computer? What with the graphics capabilities, extensive hard and software do any other home computers come anywhere near this value?

Amin Hoque, Llanrwst, Gwynedd

Early warning

For all the people who thanked me for buying a Commodore 64 earlier this year, thus ensuring a massive price drop two weeks after purchase, I am announcing that I intended to buy a 1541 disk drive in mid-October.

Undoubtedly there will be a similar price drop early in November.

You have been warned!

David Gristwood, Sunderland, Tyne & Wear

★ As a reward for your public-spiritedness, we'll be sending you our software prize

Count your pellets

I found Barbara Baldwin's Slugs and Ladders program for the TI-99/4A (HCW 23) very addictive.

I entered the program into ordinary BASIC and SAVERD it. I then LOADED it into Extended BASIC and, after a few alterations, added a display to show how many slug pellets were remaining.

I then found the program even better and faster. Here are the alterations, but remember you need Extended BASIC. An underscore represents a space.

```
131 PRINT
  "HOW
  MANY SLUG
  PELLETS?"
132 INPUT "
(20 TO 40) ":SP
133 IF (SP 20) + (SP
  40) THEN 360
  Then erase lines 350,360,
  370. Change the following
  lines:
240 CALL CHAR (112,
  "OOFFFFFFFFFFFF
  OO")
250 CALL CHAR (40,
  "10925438FE385492")
280 CALL COLOR (2,14,
  11)
340 CALL COLOR (11,
  2,16)
1700 IF K = 89 THEN
  131 ELSE 1310
```

To save your delicate Texas space bar, change line 1010 to:

```
1010 IF 46 THEN 720
  You can then use the full stop
  key to drop a slug pellet.
```

Next, insert the following lines so that the lose, win and another game request routines don't have defined characters as brackets etc.

```
1281 CALL CHARSET
1381 CALL CHARSET
1651 CALL CHARSET
1751 CALL CHARSET
```

And in Extended BASIC, the colons after a print statement should be deleted.

Michael Waldron, Sheffield, South Yorks

Keep them slaving, or you'll be sacked

At first you'll think Slave Labour is an easy game to play — but so far no-one has got past the first field. It seems to bring out the competitive spirit in everyone who plays it.

You are a task master and control a ball and chain with the control keys D and S.

A little slave runs up the screen and plants crops — for which you get a score — and then runs back down. You have to force him back up by placing the ball and chain in his path.

If you miss a slave he quickly makes his escape.

After three escapes by three different slaves you have no more chances and you're sacked from the job.

Main variables

A,B row, column number of slaves
X column of ball and chain
E ball and chain direction

Slavery has been brought back by Iain Bartram, in his novel game for the unexpanded TI-99/4A. How hard a taskmaster are you?

```
100 CALL CLEAR
110 CALL SCREEN(4)
120 CALL COLOR(1,6,4)
130 CALL CHAR(35,"3C4299A1A199423C")
140 CALL CHAR(128,"70F8F8F8740201")
150 CALL CHAR(129,"00102026494986")
160 CALL CHAR(130,"0070F8F8F8700804")
170 CALL CHAR(131,"020C1026291906")
180 CALL CHAR(132,"00387C7C7C380601")
190 CALL CHAR(133,"804040202C12120C")
200 CALL CHAR(135,"3844AA82926C206C")
210 CALL CHAR(136,"3844AA82926C6C")
220 CALL CHAR(144,"0040242428281")
221 CALL CHAR(152,"0000000000003442")
230 CALL COLOR(13,2,1)
```

```
240 CALL COLOR(14,6,1)
250 CALL COLOR(15,7,1)
251 CALL COLOR(16,15,1)
260 FOR I=2 TO 12
270 CALL COLOR(I,2,1)
280 NEXT I
290 K$="DO YOU WANT
INSTRUCTIONS?"
300 R=8
310 GOSUB 1960
320 R=10
330 K$="TO"
340 GOSUB 1960
350 R=12
360 K$="SLAVE LABOUR."
370 GOSUB 1960
380 R=14
390 K$="
"
400 GOSUB 1960
410 CALL KEY(0,K,S)
420 IF K=89 THEN 450
430 IF K=78 THEN 960
440 GOTO 410
450 CALL CLEAR
460 R=2
470 K$="RUNS ON TI-99/4A"
480 GOSUB 1960
490 R=4
500 K$="# OF IAIN BARTRAM."
510 GOSUB 1960
520 R=7
530 K$="SLAVE LABOUR"
540 GOSUB 1960
550 R=9
560 K$="YOU HAVE 3 SLAVES"
570 GOSUB 1960
580 R=11
590 K$="
"
600 GOSUB 1960
610 R=13
620 K$="TO CONTROL WITH YOUR"
630 GOSUB 1960
```

```
640 R=15
650 K$="BALL & CHAIN"
660 GOSUB 1960
670 R=17
680 K$="
"
690 GOSUB 1960
700 R=19
710 K$="KEYS D&S MOVE THE"
720 GOSUB 1960
730 R=21
740 K$="BALL & CHAIN"
750 GOSUB 1960
760 FOR D=1 TO 500
770 NEXT D
780 CALL CLEAR
790 R=2
800 K$="THE DIRECTION OF THE"
810 GOSUB 1960
820 R=4
830 K$="BALL & CHAIN ON IMPACT"
840 GOSUB 1960
850 R=6
860 K$="DETERMINES THE DIRECTION"
870 GOSUB 1960
880 R=8
890 K$="IN WHICH THE SLAVE
RETURNS TO"
900 GOSUB 1960
910 R=10
920 K$="THE FIELDS TO PLANT
MORE CROPS"
930 GOSUB 1960
940 FOR D=1 TO 400
950 NEXT D
960 CALL CLEAR
961 CALL COLOR(1,6,4)
962 FOR D=6 TO 10
963 CALL HCHAR(D,1,152,32)
964 NEXT D
970 RANDOMIZE
980 A=18
990 B=INT(20*RND)+5
```

How it works

100-120 set screen, edge colours
130-230 define characters
240-300 set colour of characters
310-460 set up title screen
470-970 instructions
980-1110 set screen for game, initialise variables
1120-1270 main subroutine for slave movement
1280-1320 ball and chain bouncing off walls
1330-1380 key control over ball and chain
1390-1410 slave bouncing off walls
1420-1690 ball and chain hitting slave back up field
1700-2020 escaping slaves routine
2030-2070 word centring for screen output
2080-2120 reset, return to main program
2130-2200 slave bouncing off walls and off ball and chain in correct direction
2210-2310 planting crop and determine direction of return
2320-2370 scoring
2380-2450 move to next field
2460-2700 end, play again?

Hints on conversion

TI BASIC is fairly standard with two main exceptions. Single statement lines are used and subroutines for graphics and sound are available, prefixed with CALL, as detailed below.
CALL CLEAR clears screen
CALL SCREEN defines screen colour
CALL CHAR defines a character with a 16-digit hex string from an eight by eight grid. Other machines will use a different method
CALL COLOR (character set, foreground, background) sets colour of character
CALL HCHAR (row, column, character code, number of repeats) places character on screen at co-ordinates specified and optionally repeats character horizontally
CALL VCHAR similar to HCHAR but repeats character vertically
CALL GCHAR (row, column, variable) similar to a screen PEEK on other machines. Gives the ASCII code at the given screen co-ordinates
CALL SOUND (D,F,V) gives sound of duration D, frequency F and volume V. Three sounds can be produced at once

Keep them slaving, or you'll be sacked

At first you'll think Slave Labour is an easy game to play — but so far no-one has got past the first field. It seems to bring out the competitive spirit in everyone who plays it.

You are a task master and control a ball and chain with the control keys D and S.

A little slave runs up the screen and plants crops — for which you get a score — and then runs back down. You have to force him back up by placing the ball and chain in his path.

If you miss a slave he quickly makes his escape.

After three escapes by three different slaves you have no more chances and you're sacked from the job.

Main variables

A,B row, column number of slaves
X column of ball and chain
E ball and chain direction

Slavery has been brought back by Iain Bartram, in his novel game for the unexpanded TI-99/4A. How hard a taskmaster are you?

```
100 CALL CLEAR
110 CALL SCREEN(4)
120 CALL COLOR(1,6,4)
130 CALL CHAR(35,"3C4299A1A199423C")
140 CALL CHAR(128,"70F8F8F8740201")
150 CALL CHAR(129,"00102026494986")
160 CALL CHAR(130,"0070F8F8F8700804")
170 CALL CHAR(131,"020C1026291906")
180 CALL CHAR(132,"00387C7C7C380601")
190 CALL CHAR(133,"804040202C12120C")
200 CALL CHAR(135,"3844AA82926C206C")
210 CALL CHAR(136,"3844AA82926C6C")
220 CALL CHAR(144,"0040242428281")
221 CALL CHAR(152,"0000000000003442")
230 CALL COLOR(13,2,1)
```

```
240 CALL COLOR(14,6,1)
250 CALL COLOR(15,7,1)
251 CALL COLOR(16,15,1)
260 FOR I=2 TO 12
270 CALL COLOR(I,2,1)
280 NEXT I
290 K$="DO YOU WANT
INSTRUCTIONS?"
300 R=8
310 GOSUB 1960
320 R=10
330 K$="TO"
340 GOSUB 1960
350 R=12
360 K$="SLAVE LABOUR."
370 GOSUB 1960
380 R=14
390 K$="
"
400 GOSUB 1960
410 CALL KEY(0,K,S)
420 IF K=89 THEN 450
430 IF K=78 THEN 960
440 GOTO 410
450 CALL CLEAR
460 R=2
470 K$="RUNS ON TI-99/4A"
480 GOSUB 1960
490 R=4
500 K$="# OF IAIN BARTRAM."
510 GOSUB 1960
520 R=7
530 K$="SLAVE LABOUR"
540 GOSUB 1960
550 R=9
560 K$="YOU HAVE 3 SLAVES"
570 GOSUB 1960
580 R=11
590 K$="
"
600 GOSUB 1960
610 R=13
620 K$="TO CONTROL WITH YOUR"
630 GOSUB 1960
```

```
640 R=15
650 K$="BALL & CHAIN"
660 GOSUB 1960
670 R=17
680 K$="
"
690 GOSUB 1960
700 R=19
710 K$="KEYS D&S MOVE THE"
720 GOSUB 1960
730 R=21
740 K$="BALL & CHAIN"
750 GOSUB 1960
760 FOR D=1 TO 500
770 NEXT D
780 CALL CLEAR
790 R=2
800 K$="THE DIRECTION OF THE"
810 GOSUB 1960
820 R=4
830 K$="BALL & CHAIN ON IMPACT"
840 GOSUB 1960
850 R=6
860 K$="DETERMINES THE DIRECTION"
870 GOSUB 1960
880 R=8
890 K$="IN WHICH THE SLAVE
RETURNS TO"
900 GOSUB 1960
910 R=10
920 K$="THE FIELDS TO PLANT
MORE CROPS"
930 GOSUB 1960
940 FOR D=1 TO 400
950 NEXT D
960 CALL CLEAR
961 CALL COLOR(1,6,4)
962 FOR D=6 TO 10
963 CALL HCHAR(D,1,152,32)
964 NEXT D
970 RANDOMIZE
980 A=18
990 B=INT(20*RND)+5
```

How it works

100-120 set screen, edge colours
130-230 define characters
240-300 set colour of characters
310-460 set up title screen
470-970 instructions
980-1110 set screen for game, initialise variables
1120-1270 main subroutine for slave movement
1280-1320 ball and chain bouncing off walls
1330-1380 key control over ball and chain
1390-1410 slave bouncing off walls
1420-1690 ball and chain hitting slave back up field
1700-2020 escaping slaves routine
2030-2070 word centring for screen output
2080-2120 reset, return to main program
2130-2200 slave bouncing off walls and off ball and chain in correct direction
2210-2310 planting crop and determine direction of return
2320-2370 scoring
2380-2450 move to next field
2460-2700 end, play again?

Hints on conversion

TI BASIC is fairly standard with two main exceptions. Single statement lines are used and subroutines for graphics and sound are available, prefixed with CALL, as detailed below.
CALL CLEAR clears screen
CALL SCREEN defines screen colour
CALL CHAR defines a character with a 16-digit hex string from an eight by eight grid. Other machines will use a different method
CALL COLOR (character set, foreground, background) sets colour of character
CALL HCHAR (row, column, character code, number of repeats) places character on screen at co-ordinates specified and optionally repeats character horizontally
CALL VCHAR similar to HCHAR but repeats character vertically
CALL GCHAR (row, column, variable) similar to a screen PEEK on other machines. Gives the ASCII code at the given screen co-ordinates
CALL SOUND (D,F,V) gives sound of duration D, frequency F and volume V. Three sounds can be produced at once

TI-99/4A PROGRAM

```

1000 X=15
1010 ADIR=-1
1020 IF RND>.5 THEN 1050 ELSE 1030
1030 BDIR=-1
1040 GOTO 1060
1050 BDIR=1
1060 CALL KEY(0,K,S)
1070 CALL HCHAR(20,X,32,2)
1080 IF K=68 THEN 1260
1090 IF K=83 THEN 1290
1100 CALL HCHAR(20,X,128)
1110 CALL HCHAR(20,X+1,129)
1120 IF (X>29)+(X<3) THEN 1210
1130 CALL HCHAR(A,B,Q)
1140 A=A+ADIR
1150 B=B+BDIR
1151 CALL GCHAR(A,B,Q)
1160 CALL HCHAR(A,B,135)
1170 IF A<11 THEN 2120
1180 IF (B<2)+(B>31) THEN 1320
1190 IF A>18 THEN 1350
1200 GOTO 1060
1210 E=-E
1220 X=X+E
1230 CALL HCHAR(20,X-E,32,2)
1240 CALL SOUND(30,300,2)
1250 GOTO 1100
1260 E=2
1270 X=X+E
1280 GOTO 1100
1290 E=-2
1300 X=X+E
1310 GOTO 1100
1320 BDIR=-BDIR
1330 CALL SOUND(40,350,1)
1340 GOTO 1190
1350 IF B=X THEN 1360 ELSE 1490
1360 CALL HCHAR(20,X,32,2)
1370 CALL HCHAR(19,X,132)
1380 CALL HCHAR(20,X+1,133)
1390 FOR D=1 TO 15
1400 NEXT D
1410 B=B+1
1420 CALL HCHAR(A,B,135)
1430 FOR D=1 TO 35
1440 NEXT D
1450 CALL HCHAR(19,X,32)
1460 CALL HCHAR(20,X,128)
1470 CALL HCHAR(20,X+1,129)
1480 GOTO 2060
1490 IF B=X+1 THEN 1500 ELSE 1630
1500 CALL HCHAR(20,X,32,2)
1510 CALL HCHAR(19,X+1,130)
1520 CALL HCHAR(20,X+1,131)
1530 FOR D=1 TO 15
1540 NEXT D
1550 B=B+1
1560 CALL HCHAR(A,B,135)
1570 FOR D=1 TO 35
1580 NEXT D
1590 CALL HCHAR(19,X+1,32)
1600 CALL HCHAR(20,X,128)
1610 CALL HCHAR(20,X+1,129)
1620 GOTO 2060
1630 K$="ESCAPED SLAVES "
1640 R=23
1650 GOSUB 1960
1660 SL=SL+1
1670 GOSUB 1690
1680 GOTO 1770
1690 FOR M=1 TO 2
1700 CALL HCHAR(A,B,32)
1710 A=A+1
1720 CALL HCHAR(A,B,135)
1730 FOR D=1 TO 10
1740 NEXT D

```

```

1750 NEXT M
1760 RETURN
1770 C=135
1780 FOR M=B TO 31
1790 CALL HCHAR(A,B,32)
1800 B=B+1
1810 C=271-C
1820 CALL HCHAR(A,B,C)
1830 FOR D=1 TO 15
1840 NEXT D
1850 NEXT M
1860 GOSUB 1690
1870 FOR M=25+SL TO 32
1880 CALL HCHAR(A,B,32)
1890 B=B+1
1900 C=271-C
1910 CALL HCHAR(A,B,C)
1920 FOR D=1 TO 15
1930 NEXT D
1940 NEXT M
1950 GOTO 2010
1960 W=16-INT(LEN(K$)/2)
1970 FOR I=1 TO LEN(K$)
1980 CALL HCHAR(R,W+I,ASC(SEG$(K$,I,1)))

```



This week's issue of Home Computing Weekly features a sample sheet from Computer Agencies' Pixel Pad for the Spectrum



For a pad of 50 sheets send just £4.99, including post and packing, to:

BROWN COMPUTING & CO
29 Sun Street, Hitchin, Herts
SG5 1AT

Delivery will be within 14 days

Trade enquiries to: Robert Gibson & Co (0799) 41424

TI-99/4A PROGRAM

```

380 CALL CHAR(128,"FFFFFFFFFFFF
FFF")
390 CALL CHAR(136,"0040000090000
080")
400 CALL CHAR(144,"3C7E010F42358
118")
410 CALL CHAR(152,"FFFFFFFFFFFF
FFF")
390 CALL CHAR(136,"0040000090000
080")
400 CALL CHAR(144,"3C7E010F42358
118")
410 CALL CHAR(152,"FFFFFFFFFFFF
FFF")
420 CALL HCHAR(1,1,136,(24*32))
430 CALL HCHAR(23,1,128,64)
440 CALL HCHAR(22,7,128,19)
450 FOR J=11 TO 20 STEP 3
460 CALL HCHAR(21,J,73)
470 NEXT J
480 CALL HCHAR(21,7,73)
490 CALL HCHAR(21,25,73)
500 CALL HCHAR(1,1,128,128)
510 CALL VCHAR(1,1,128,72)
520 CALL VCHAR(1,30,128,72)
530 CALL COLOR(2,16,2)
540 CALL COLOR(3,2,16)
550 CALL COLOR(4,2,16)
560 CALL COLOR(5,16,2)
570 CALL COLOR(6,16,2)
580 CALL COLOR(7,16,2)
590 CALL COLOR(8,16,2)
600 CALL COLOR(13,15,1)
610 CALL COLOR(14,2,2)
620 CALL COLOR(15,12,1)
630 CALL COLOR(16,7,1)
640 CALL SCREEN(6)
650 M$="ASTRO-GALLERY"
660 E=23
670 F=9
680 GOSUB 960
690 M$="****POINTS****"
700 E=1
710 F=8
720 GOSUB 960
730 M$="PLUS"
740 E=1
750 F=3
760 GOSUB 960
770 M$="MINUS"
780 E=1
790 F=24
800 GOSUB 960
810 M$="SHOTS***"&B$
820 E=3
830 F=12
840 GOSUB 960
850 HSC$=STR$(HSC)
860 M$="H/SC**"&HSC$
870 E=24
880 F=1
890 GOSUB 960
900 U$=STR$(U)
910 M$="AMMO**"&U$
920 E=24
930 F=23
940 GOSUB 960
950 GOTO 1000
960 FOR J=1 TO LEN(M$)
970 CALL HCHAR(E,F+J,ASC(SEG$(M$,J,1)))
980 NEXT J
990 RETURN
1000 CALL VCHAR(13,31,79,10)

```

```

1010 GOSUB 2030
1020 GOSUB 1900
1030 R=12
1040 R2=12
1150 IF N<>6 THEN 1200
1160 P=P-1
1170 CALL HCHAR(P,31,152)
1180 IF P=13 THEN 1610
1190 N=0
1200 CALL SOUND(-100,110,3,1047,
3,1760,3)
1210 ON X GOTO 1220,1240,1260,12
80
1220 R=R+1
1230 GOTO 1290
1240 R=R-1
1250 GOTO 1290
1260 C=C+1
1270 GOTO 1290
1280 C=C-1
1290 CALL HCHAR(V,W,136)
1300 IF R<5 THEN 1220
1310 IF R>20 THEN 1240
1320 IF C>29 THEN 1280
1330 IF C<4 THEN 1260
1340 GOSUB 1640
1350 IF B>U THEN 1360 ELSE 1380
1360 E$="GAME OVER - GAME OVER!"
1370 GOTO 2100
1380 X=INT(RND*4)+1
1390 CALL SOUND(-100,554,3,1047,
3,1760,3)
1400 ON X GOTO 1410,1430,1450,14
70
1410 R2=R2+1
1420 GOTO 1480
1430 R2=R2-1
1440 GOTO 1480
1450 C2=C2+1
1460 GOTO 1480
1470 C2=C2-1
1480 CALL HCHAR(V2,W2,136)
1490 IF R2<5 THEN 1410
1500 IF R2>20 THEN 1430
1510 IF C2>29 THEN 1470
1520 IF C2<4 THEN 1450
1530 GOSUB 1640
1540 IF M<>1 THEN 1600
1550 B$=STR$(B)
1560 FOR J=1 TO LEN(B$)
1570 CALL HCHAR(3,20+J,ASC(SEG$(B$,J,1)))
1580 NEXT J
1590 M=0
1600 IF B>U THEN 1610 ELSE 1630
1610 E$="GAME OVER - GAME OVER!"
1620 GOTO 2100
1630 GOTO 1070
1640 CALL KEY(0,K,S)
1650 IF S=0 THEN 1950
1660 IF (K<49)+(K>54) THEN 1950
1670 K=K-48
1680 ON K GOTO 1690,1710,1730,17
50,1770,1790
1690 CQ=7
1700 GOTO 1800
1710 CQ=11
1720 GOTO 1800
1730 CQ=14
1740 GOTO 1800
1750 CQ=17
1760 GOTO 1800
1770 CQ=20
1780 GOTO 1800
1790 CQ=25

```

```

1800 CALL SOUND(-150,-6,0)
1810 CALL VCHAR(7,CQ,46,14)
1820 CALL VCHAR(7,CQ,136,14)
1830 B=B+1
1840 M=1
1850 IF (CQ=C)+(CQ=C2) THEN 1860
ELSE 1880
1860 GOSUB 1960
1870 RETURN
1880 L=L+A
1890 CALL SOUND(-250,110,0)
1900 M$=STR$(L)
1910 FOR J=1 TO LEN(M$)
1920 CALL HCHAR(3,26+J,ASC(SEG$(M$,J,1)))
1930 NEXT J
1940 IF (L>SC)+(L)=3010 THEN 209
0
1950 RETURN
1960 SC=SC+100
1970 CALL SOUND(-500,523,0,659,0
,784,0)
1980 CALL COLOR(14,7,7)
1990 CALL SCREEN(7)
2000 CALL COLOR(14,12,13)
2010 CALL COLOR(14,7,12)
2020 CALL SCREEN(6)
2030 M$=STR$(SC)
2040 FOR J=1 TO LEN(M$)
2050 CALL HCHAR(3,3+J,ASC(SEG$(M$,J,1)))
2060 NEXT J
2070 CALL COLOR(14,2,2)
2080 RETURN
2090 E$="THE GALLERY HAS BEATEN
YOU!"
2100 FOR DELAY=1 TO 1500
2110 NEXT DELAY
2120 CALL CLEAR
2130 FOR SET=1 TO 8
2140 CALL COLOR(SET,14,15)
2150 NEXT SET
2160 SC=((SC-((U-B)*5))-L)*T
2170 IF SC<0 THEN 2180 ELSE 2190
2180 SC=0
2190 IF SC>HSC THEN 2200 ELSE 22
10
2200 HSC=SC
2210 CALL SCREEN(15)
2220 PRINT E$::
2230 PRINT "YOUR FINAL SCORE":
ADJUSTED FOR DIFFICULTY": "IS:
::
2240 PRINT SC::
2250 PRINT "BEST SCORE SO FAR:
::
2260 PRINT HSC::
2270 PRINT "PRESS ANY KEY TO PLA
Y AGAIN"
2280 CALL KEY(0,K,S)
2290 IF S=0 THEN 2280
2300 L=0
2310 N=0
2320 GOTO 200

```


LETTERS

Electron by joystick

I thought other TI owners would like to know about a conversion I have made, which allows the Electron game in HCW 28 to be played with joysticks. This may also be used to create general-purpose joystick subroutines — something I found hard at first.

My conversion also includes changes to CALL KEY statements, so that they will accept lower case letters as well as upper case.

```
860 CALL JOYST(I,X,Y)
870 (ENTER)
880 IF X=4 THEN 885
    ELSE 890
885 IF Y=0 THEN 930
    ELSE 890
890 IF X=0 THEN 895
    ELSE 900
895 IF Y=-4 THEN 1020
    ELSE 900
900 IF X=0 THEN 905
    ELSE 910
905 IF Y=4 THEN 1110
    ELSE 910
910 IF X=-4 THEN 915
    ELSE 860
915 IF Y=0 THEN 1200
    ELSE 860
1840 CALL KEY(0,K,S)
1860 IF K=121 THEN 100
2470 IF K=121 THEN 2500
2480 IF K=110 THEN 2490
    ELSE 2420
2930 CALL KEY(0,K,S)
2950 IF K=114 THEN 2500
```

I also thought you'd like to know my high-score on Parsec — 309,200. It would be interesting if other TI owners who read your magazine and own Parsec could write in with the high scores that they have got on this very hard game.

By the way, on a recent holiday to Corfu I was surprised to find a copy of HCW on sale in a newsagents there!

R. C. Sanderson,
Rotherham, South Yorks

The bubble explodes

We were utterly amazed to read your review on our program called Hustler, a pool game for the Commodore 64. At the time we read it we had just sold out of them at the PCW show at the Barbican.

Your review mentions the program hanging up, and highlights various other defects. Unfortunately you have reviewed an early development copy which bears little resemblance to the finished article.

We would deem it a favour if you would put the record straight for your readers.

Mark Meakings, Bubble Bus, Tonbridge, Kent

Send your letter to Letters, Home Computing Weekly, 145 Charing Cross Road, London WC2H 0EE. Don't forget to name your computer — the best letter could win £5-worth of software. Got a problem with your micro? We'll soon be starting a queries page, so send your technical questions to Queries at the above address



Missing symbols

I was pleased to see my COMX 35 program in HCW 31, but some of the program seems to have got lost along the way!

In case any readers were puzzled by the program, the missing bits are:

```
230 between A and 3 should
    be a less than sign
236 between B and 11 should
    be a less than sign
232 between A and 20 a
    greater than sign
234 between B and 28 a
    greater than sign
240 the character between
    quotes is shift S
```

Deborah Horne

Checking out Checkers

I understand that some people have been having difficulties with my Micro Checkers program, published in HCW 31.

The program will run as listed, provided that the CTRL key is held down while the positions are chosen. This ensures that moves have to be made deliberately.

If a line 1105 is added;
1105 IF (R<4)+(R>19) THEN 1070
and line 1170 is altered to
1170 IF (C<8)+(C>23) THEN 1130
then keys pressed without the control button being held down will have no effect, and it will be impossible to pick a square off the board.

The program works in this way because lines 1070 and 1130 use key unit 4, in which control codes 1 to 31 are active. This way, the values returned when a key is pressed translate directly into positions on the board.

If you want to avoid having to press the CTRL key, then you should alter line 1170 to read

```
1170 IF (R<69)+(R>83)
    +(C<72)+(C>86) THEN
    1210
```

In this way, the program

will work in its original form. If the line value 1210 is changed to 1070, the choice of squares off the board will be prevented.

If you choose to amend the program in this way you will also need to add two more lines

```
1171 R=R-64
```

```
1172 C=C-64
```

Lance Booth, Burton-on-Trent, Staffs

Invader: Arcadia's answer

I was pleased to read the letter titled Mushroom Mix in your No. 26 edition. It gave me a ray of hope regarding Oric Invader which I ordered in March 1983 from Arcadia.

After reading Mr Love's letter regarding Arcadia's problems, I wrote to Mr Love explaining that I sent a cheque for Oric Invader, that the cheque had been cashed and that I have not received the cassette yet.

After waiting one month for a reply from Mr Love (which seems to be normal reply time for computer companies), I thought it was time I wrote to Home Computing Weekly to let other readers know what they are letting themselves in for.

Mr Love's way of dealing with queries seems to be to forget them.

After reading several computer mags, my problem seems to be the norm. Is this the way British firms get their working capital?

M. Schofield, Wakefield West Yorks

Mr Love, of Arcadia, said "Oh dear! That must have been one of the first orders we received for Oric Invader. I'll make sure Mr Schofield gets another tape. We did experience delivery delays when we switched duplicating companies due to loading problems, but now we've delivery down to within a week of receiving the customer's order. Our tapes are also being distributed to some retail stores now."

Is this a record?

In HCW 24 you had an article about Pasesoft which said that Tony Reid had achieved a score of 124,000 on TI Parsec.

I have beaten this score easily with a score of 1,080,500 on my TI-99/4A.

Has any other reader of HCW beaten this score? If not, is this an all-time high?

Paul Hopkins, Downend Bristol

BUYERS' GUIDE

TI-99/4A

£99.95

16K RAM (14.5K useable) 26K ROM

Keyboard: typewriter style, 48 keys, including row of function keys (not user-definable), space bar. Two functions per key. **Screen:** 32 columns by 24 rows text, 128 characters, all user-definable, 16 colours each for foreground and background. The way colour is handled is rather odd — a colour is assigned to a particular group of characters. There are 16 groups of eight characters, so a particular colour can be assigned, for example, to the letter A but will also affect B to H. High-resolution graphics only available with TI plug-in cartridges. **Sound:** three channels of five octaves with noise channel.

Languages: TI BASIC is included, but cannot use machine code, so arcade quality programs cannot be written or bought on cassette for the standard model. To write or use machine code programs, TI's Mini Memory or Editor Assembler cartridges are needed. Machine code can also be used with Extended BASIC, provided 32K is added. LOGO, the teaching language, is also available. **Processor:** TMS 9900. **Operating system:** custom. **I/O:** joystick and cartridge ports. **Number sold in UK:** TI would not comment, our estimate: 50,000. **Speed:** slow.

Add-ons: Apart from joysticks and speech synthesiser, all need a peripheral expansion box. Available: disc drives, 32K RAM card, RS232 serial port for printers. TI's own cassette recorder includes a cable, your own will need a special cable. There are very few independent add-ons.

Software: More software is becoming available — including US imports — but you may be forced to buy through mail order. TI sells a range of sophisticated cartridge software which is more expensive.

Opinion: an interesting computer which is now quite good value at its new low price. And it looks like a real machine with its good keyboard. Its drawback is that machine code cannot be used with the standard model so there is not much arcade-type cassette software, although this is slowly increasing. However, if all the games that interest you are from TI and you are willing to pay cartridge prices, then this is a good machine. For programmers it has a line editor which, although one of the better versions, is awkward to use. The BASIC is slow but easy to use and comprehensive. Its worst features are that there can be just one statement per line and spaces are essential. Maths functions are good but string functions leave something to be desired. The computer is very well built and looks attractive.

Texas Instruments, Manton Lane, Bedford MK41 7PA
Bedford (0234) 67466

Tandy MC-10

£99.95

4K RAM 8K ROM

Keyboard: 47 hard plastic keys in Spectrum-style layout, includes 16 graphics keys, space bar. Single key keyword entry. **Screen:** 32 columns by 16 lines text, 64 by 32 graphics. Eight colours.

Sound: single channel, five octaves controlled from BASIC. **Languages:** Microsoft BASIC. **Processor:** 6803. **Operating system:** Microsoft. **I/O:** RS-232. **Number sold in UK:** no comment from Tandy, but small because computer launched recently. **Speed:** medium to fast.

Add-ons: 16K RAM pack. Also compatible with Tandy printers.

Software: a small range, but growing.

Opinion: Tandy is selling this computer through 220 of its stores, rising to around 300, so support is excellent. It has been on sale in the US since May, so imported software from independents is likely. No UK independent software is known. You may feel the relatively high price is partly offset by Tandy's reputation for service and advice.

Tandy, Tameway Tower, Bridge Street, Walsall, West Midlands WS1 1LA
0922 648181



Spectrum 48K

£129.95

48K RAM (39.5K useable) 16K ROM

Identical to 16K version in every respect except price and memory and the fact that more software is available due to increased memory size.

Sinclair Research, Camberley, Surrey GU15 3BR
Camberley (0276) 685311

Oric 48K

£139.95

48K RAM (46K useable in text mode, 39.5K free in graphics mode) 16K ROM

Identical to the 16K version except for the memory differences and certain control lines on the expansion port to allow the ROM to be switched out and more memory used.

Oric Products International, Coworth Park, London Road, Ascot, Berks SL5 7SE
Ascot (0990) 27686

VIC-20

£139.95

6K RAM (3.5K useable) 20K ROM

Keyboard: typewriter quality, 67 keys, space bar, dedicated cursor controls, four user definable function keys.

Screen: 22 columns by 23 lines, eight colours for text, 16 for background and border and graphics. All characters user definable, lower case, inverted output, two character sets. High-resolution possible with expansion cartridge or with extra programming but BASIC does not support this. Two modes, 160 by 128 in eight colours and 80 by 128 in 16 colours.

Sound: three channels and one noise channel covering nine octaves through TV speaker.

Language: Commodore BASIC supplied, compatible with PET and Commodore 64, Extended BASIC, at extra cost gives access to the hi-res, sound, and user definable function keys. Forth, Comal and Pilot are available. **Processor:** 6502A. **Operating System:** Commodore Kernal. **I/O:** IEEE 488 serial bus, RS232, user port, cartridge port, joystick port, two paddle port (two analogue to digital converters). **Number sold in UK:** 500,000.

Speed: medium to fast.

Add-ons: lots available due to its immense popularity. From Commodore alone there are printers, plotters, disk drives, modems and, via an IEEE 488 parallel cartridge, access to Commodore, Hewlett Packard and laboratory instruments. Disc drives up to a hard disk Winchester holding 32 Megabytes. The computer is sold as a package including the Commodore cassette recorder which is needed to record programs on tape. Memory expansion up to 24K is available.

Software: Many cassette tapes and cartridges from independents and Commodore.

Opinion: this is a nice machine to get started on and it comes as a package containing a computer, a cassette recorder, a beginners guide to BASIC and a games tape. So the initial high price for a 3.5K machine is somewhat offset. The BASIC is very easy to learn and is Microsoft compatible. This type of BASIC was first used by Commodore on the first widely available micro in 1977 so there is a lot of compatible software around.

Commodore Business Machines, 675 Ajax Avenue, Slough, Berks SL1 4BG
Slough (75) 79292/74111

Girls and micros

I am researching the differences, if any, between males and females in computer skills and interests, particularly with regard to computer games, and I would welcome assistance from your readers.

The information can be given anonymously if desired but I need to know the sex of the correspondents, approximately how many hours each week are spent using the computer, what percentage of the time is spent playing games and the names of their three favourite games with the relevant highest scores if appropriate.

I would also be interested to know of any other work your readers know of in similar areas, and would be very pleased to hear their views on why so few girls and women seem to be interested in micros.

Anne Shadbolt, Sheffield City Polytechnic, Totley Hall Lane, Sheffield S17 4AB

Chatter talk

We were very pleased to see quite favourable reviews of our programs, Chattermaths and Chattercount (HCW 34).

We must assume that, for some reason, the reviewer had not been given a copy of the instructions.

These are not within the program because of insufficient memory and the age of the children playing. They should be on the cassette insert.

Could we therefore make the following points?

Chattermaths includes subtraction as well as addition, and has two levels of difficulty: answers up to 10 and answers up to 10+10.

There is a simple form of scoring which can be reset to zero for use by another child.

In Chattercount, the 'mediocre' number recognition routine is intentionally simple, and acts only as an introduction to the numbers on the keyboard.

A child who gets five correct answers automatically moves onto a slightly harder version of the same game. Spoken help is given when necessary in each game.

There is a choice between numbers up to five and numbers up to nine.

These programs are also among those available for use on the Spectrum with the Currah Microspeech.

B. M. & S. M. Hodgson, Gilling East, Yorks.

Can we put in a plea here to software companies to ensure that the review tapes you send us are the versions that will be on sale, complete with final packaging and instructions? Our reviewers can't comment on what they haven't been sent, so it's in your interests to make sure they get the finished product.

Computing in Clifton?

I am a TRS-80 III and Spectrum user, and I do not know much about programming. I am looking for some people who live in Bristol (preferably Clifton) who knows how to program such computers as ZX81, Spectrum, VIC-20, Dragon, TRS-80 III, Oric and BBC, to help me start a club.

This club will mainly deal in games and maybe a few business programs, and I hope that as we go along we will be able to open a software library. If you are interested please, please contact me at the address below. Christopher Tihanyi, Basement Flat, 2 Victoria Sq, Clifton, Bristol 8

Don't just blame Rabbit

To begin with, I must say I totally agree with T.M.'s review (HCW 33) of the VIC-20 game, Paratroopers.

I know I am right in saying that it is nobody's idea of fun to slaughter defenceless figures and watch and enjoy their descent to the ground, even if they are only made-up characters.

But I do think it is a little unfair for us to make Rabbit (the company producing Paratroopers) the only scapegoat for this "blow 'em all up and get the points" type of game.

After all, when you come to think of it, aren't nearly all the games out nowadays connected in some way to "killing"? Whether it's zapping, munching, flattening or just disintegrating, it's still the same, isn't it?

Perhaps this is a worse example than most, but it's an unavoidable fact that you are committing the accused "sin" when you play games like Scramble, Galaxion and so on.

On the same page there was another game very similar to this one — only you bomb ships instead — entitled Harrier Attack,

which received four stars.

I am not saying I approve or disapprove of these games. I am merely trying to emphasise that Paratroopers is not the only one of its kind.

Secondly, I think the children of today have enough to worry about — what with the wars we hear about every day — without computer tapes of this kind.

Finally, I would like to say that in my opinion HCW is, minus the odd thing like this, the best computer magazine on the ever-expanding home computing book shelf. Keep it up!

Luke Jacobson, aged 12, London SW15

★ Software winner

Life, death and micro games

Now that I have had time to get over T.M.'s review of "Paratroopers" in the October 18th issue I am able to see its funny side.

In fact, I can even compliment the reviewer on the first four lines of his review. By saying "This is one of the best examples of user-defined graphics that I have ever seen, along with excellent sound effects" he reflects accurately some of the best qualities of the game.

It is after this that he seems to lose sight of them by indulging himself in his own curiously narrow ideas of morality.

Confrontation, life and death are the "meat and drink" of computer games. No one takes them seriously, or the idea of humans being involved.

On the same page as the "Paratroopers" review are four others involving humans. Three feature confrontations which are exclusively human, and one is "Harrier Attack" which to some people might bring back memories of a real human conflict.

So, why is T.M. the only reviewer to feel that this kind of game is immoral? The game players don't seem to share his views either, because "Paratroopers" is already a best seller.

A reviewer has a responsibility, both to the game manufacturers and whoever reads his review, which should take precedence over his own personal views. He is there to evaluate each game on its technical merits, and not to treat readers to his feelings about Martians, monsters, helicopters or anything else.

In this case responsibility seems to have been shot down in favour of prejudice and self indulgence. How could any game, however dire, be worth only 0% for playability and value for money? Is this a trend? Can we expect to see "Phoenix" reviewed by the RSPCA in your next issue? And "Planetoids" reviewed by the UFO Society? Or, is it just a one-off?

Heather Lamont, Rabbit Software, Harrow, Middlesex

Not so private lines - 1

On the Micro Tips page of HCW 31, Andrew Turek gave a tip called Private Lines which he says makes it impossible to break into a program without a code word.

I entered his sequence into one of my Data programs, following his instructions carefully.

I did indeed find that entering anything other than the code word would not gain access, but being a fairly pessimistic sort, I looked for other ways to break in and eventually found one.

If you type a false code word, then press ENTER, and very quickly SHIFT and BREAK together you get LBREAK. From there, ENTER or LIST will let you into the program.

Ron Rickwood, Gillingham, Kent

Not so private lines - 2

Nice try, Andrew Turek (Private Lines, HCW31), but not quite good enough. By using SHIFT 6 you can get a Report H STOP in INPUT. However, by disabling the error report by adding a line POKE 23613, 215 you can make the program very inaccessible without the keyword.

Here is how I would use Andrew's program, not forgetting to reset the ERR SP back to normal in the first line of your program with POKE 23616,84

```
1 POKE 23613,84
10 PRINT AT 10,5;"YOUR SECRET PROGRAM"
30 PAUSE 0
9995 POKE 23613,255
9996 FOR c=1 TO 3:CLS:PRINT "access keyword please?":INPUT LINE x$:IF x$=s$ THEN GO
```

TO 999 8
9997 NEXT c:NEW
9998 CLS:GO TO 1
9999 SAVE "secret" LINE 9995

Remember to enter your keyword as a direct command, using LET s\$="word or symbol" then SAVE by GOTO 9999.

Alan Gee, Grantham, Lincs.

Take ten lines...

A few lines seem to have been missed out of my Astro Gallery game for the TI-99/4A, published in HCW 35. The missing lines are:

```
1050 C=20
1060 C2=10
1070 CALL HCHAR(R,C,144)
1080 CALL HCHAR(R2,C2,144)
1090 V=R
1100 V2=R2
1110 W=C
1120 W2=C2
1130 X=INT(RND*4)+1
1140 N=N+1
```

Tony Garner

...And one more

Someone must have spirited away line 40110 from my Halloween adventure in HCW 31. It should read: 40110 DATA KITCHEN, 0,2,0,0

Alan Blackham

Tip typo

Thank you for publishing my tip for finding memory locations on the VIC-20 in HCW. However, I have found a small typing error crept in, for which I must apologise. The amended line is below.

```
LN FOR CH=0 TO CN:
  READ CS: C=(ASC(CS)-64)*8+7168
LN FOR CS=0 TO 7:
  READ CQ: POKE C+CS,CQ: NEXT CS,CH
```

where LN=line number(s) and CN the number of characters to define.

Also, in a long program, it is better to split the original line into two as shown.

Again the matching DATA line would be as: IN DATA "X",123,234,0,34,62,1,3,0

David Shepherdson, Ilkley, West Yorks

How it works

85-95 variables set-up
100 clears screen
105 sets screen colour to black
110-120 sets text to white
125-150 define colours
155-185 define characters
186 do you want instructions?
190-205 displays screen border
210-265 displays orchard
270-320 displays random positions of apples
325-370 displays random position of key
375 displays man
380-525 main game loop
380-520 count down farmer's movement along rows
385-390 & 510 control farmer's movement along columns
395 checks keyboard
400 clears old player from screen
405-440 check for movement
445-470 check new screen positions
475 displays new player
480 adds one to time limit
485 tests to see if time limit is exceeded
490-495 check new farmer position
500-505 move farmer
515 displays basket
530-540 you fell over a basket
545-555 you touched the electric fence
560-570 you picked up an apple
575-605 you picked up the key
610-615 you have run out of time
620-650 the farmer has caught you
655-690 you escaped
705-725 losing tune
745-755 displays message
760 print score
770 end
805-895 display instructions
915-925 type in time limit

An apple a day might keep the doctor away, but what about the farmer that owns the orchard?

As you run around the orchard picking up apples, he's on your trail, picking up the fruit you've missed and dropping baskets to block your escape.

If he catches up with you, he'll prosecute you for trespassing on his land.

Main variables

X column of figure
Y row of figure
SC score
T/U positions of apples
GB number of apples
B/P position of key
R/C movement directions for farmer
M positions for baskets
LIMIT time limit
K position for gate
CS screen messages

You move by using the four arrow keys with the alpha lock button down. Mind you don't trip over a basket, or touch the edge of the orchard — it's protected by an electric fence.

When you've collected the apples, the only way out of the orchard is to pick up the key

Pip the farmer to the post

Try to escape with the fruit before that farmer catches up with you. Stuart Lenker explains how to play his apple-picking game for the standard TI-99/4A

```
10 REM *****APPLE SCRUNGING*****
20 REM *****STUART LENKER*****
30 REM *****AUGUST 1983*****
85 Y=22
90 X=14
95 SC=0
100 CALL CLEAR
105 CALL SCREEN(2)
110 FOR I=1 TO 8
115 CALL COLOR(1,16,1)
120 NEXT I
125 CALL COLOR(11,15,1)
130 CALL COLOR(12,13,1)
135 CALL COLOR(13,7,1)
140 CALL COLOR(14,11,1)
145 CALL COLOR(15,16,1)
150 CALL COLOR(16,8,1)
155 CALL CHAR(119,"3C4281FFFFFFFF7E")
160 CALL CHAR(125,"FFFFFFFFFFFFFFFF")
165 CALL CHAR(130,"8304053C7E7E7E3C")
170 CALL CHAR(140,"3826381010301070")
175 CALL CHAR(145,"10103C5A5A10103C")
180 CALL CHAR(155,"6562F2FEF26262F2")
185 CALL CHAR(156,"FFFFFFFFFFFFFFFF")
186 INPUT "DO YOU WANT INSTRUCTIONS?" ; Z
187 CALL CLEAR
188 IF Z="Y" THEN 805 ELSE 915
190 CALL HCHAR(1,1,125,27)
195 CALL HCHAR(23,1,125,27)
200 CALL VCHAR(1,1,125,23)
205 CALL VCHAR(1,27,125,23)
210 READ A,B,C
215 IF A=0 THEN 270
220 CALL HCHAR(A,B,125,C)
225 GOTO 210
230 DATA 3,3,5,4,3,5,5,3,5,6,3,5,8,3,5,9,3,5,10,3,5,11,3,5
235 DATA 13,3,5,14,3,5,15,3,5,16,3,5,18,3,5,19,3,5,20,3,5,21,3,5
240 DATA 3,9,5,4,9,5,5,9,5,6,9,5,8,9,5,9,5,10,9,5,11,9,5
245 DATA 13,9,5,14,9,5,15,9,5,16,9,5,18,9,5,19,9,5,20,9,5,21,9,5
250 DATA 3,15,5,4,15,5,5,15,5,6,15,5,8,15,5,9,15,5,10,15,5,11,15,5
255 DATA 13,15,5,14,15,5,15,15,5,16,15,5,18,15,5,19,15,5,20,15,5,21,15,5
260 DATA 3,21,5,4,21,5,5,21,5,6,21,5,8,21,5,9,21,5,10,21,5,11,21,5
265 DATA 13,21,5,14,21,5,15,21,5,16,21,5,18,21,5,19,21,5,20,21,5,21,21,5,0,0,0
270 T=-3
275 T=T+5
280 IF T=27 THEN 325
285 GB=0
290 GB=GB+1
295 IF GB>4 THEN 320
300 RANDOMIZE
305 U=INT(248RND)+2
310 CALL HCHAR(T,U,130)
315 GOTO 290
320 GOTO 275
325 V=INT(38RND)+1
330 B=INT(20RND)+3
335 IF V=1 THEN 355
340 IF V=2 THEN 365
345 P=20
350 GOTO 370
355 P=8
360 GOTO 370
365 P=14
370 CALL HCHAR(B,P,140)
375 CALL HCHAR(Y,X,145)
380 FOR R=2 TO 22 STEP 5
385 M=INT(248RND)+2
390 FOR C=2 TO M
395 CALL KEY(0,K,S)
400 CALL HCHAR(Y,X,32)
405 IF K<83 THEN 415
410 X=X-1
415 IF K<68 THEN 425
420 X=X+1
425 IF K<69 THEN 435
430 Y=Y-1
435 IF K<88 THEN 445
440 Y=Y+1
445 CALL GCHAR(Y,X,GET)
446 CALL SOUND(4,-5,10)
450 IF GET=119 THEN 530
455 IF GET=125 THEN 545
460 IF GET=130 THEN 560
465 IF GET=140 THEN 575
470 IF GET=156 THEN 655
475 CALL HCHAR(Y,X,145)
480 TIME=TIME+1
485 IF TIME>LIMIT THEN 610
490 CALL GCHAR(R,C,GET)
495 IF GET=145 THEN 620
500 CALL HCHAR(R,C,155)
505 CALL HCHAR(R,C,32)
510 NEXT C
515 CALL HCHAR(R,C,119)
520 NEXT R
```

```
525 GOTO 380
530 CALL SOUND(40,-5,0)
535 C$="WHOOPS-YOU FELL OVER A BASKET!"
540 GOTO 695
545 CALL SOUND(500,-7,3)
550 C$="YOU TOUCHED THE ELECTRIC FENCE!"
555 GOTO 695
560 CALL SOUND(300,1000,0)
565 SC=SC+10
570 GOTO 475
575 FOR I=1 TO 8
580 CALL SOUND(100,1000,0)
585 NEXT I
590 SC=SC+50
595 K=INT(248RND)+3
600 CALL HCHAR(1,K,156)
605 GOTO 475
610 C$="BAD LUCK-YOU'VE RUN OUT OF TIME!"
615 GOTO 695
620 CALL SOUND(150,196,0)
625 CALL SOUND(300,247,0)
630 CALL SOUND(150,196,0)
635 CALL SOUND(150,220,0)
640 CALL SOUND(300,196,0)
645 C$="FARMER McDONALD HAS CAUGHT YOU!"
650 GOTO 695
655 CALL SOUND(100,440,0)
660 CALL SOUND(100,550,0)
665 CALL SOUND(100,440,0)
670 CALL SOUND(100,660,0)
675 CALL SOUND(100,880,0)
680 C$="WELL DONE-YOU DIDN'T GET CAUGHT!"
685 SC=SC*2
690 GOTO 730
695 FOR DELAY=1 TO 300
```

```
700 NEXT DELAY
705 TONE=1100
710 FOR C=1 TO 10
715 CALL SOUND(200,TONE,0)
720 TONE=TONE-110
725 NEXT C
730 FOR DELAY=1 TO 500
735 NEXT DELAY
740 CALL CLEAR
745 FOR I=1 TO LEN(C$)
750 CALL HCHAR(B,I,ASC(SEG(C$,I,1)))
755 NEXT I
760 PRINT "YOUR SCORE WAS";SC
765 FOR DELAY=1 TO 1000
770 NEXT DELAY
775 CALL CLEAR
780 END
805 PRINT "MOVE" ; " " ; "AROUND THE ORCHARD,"
810 CALL HCHAR(23,8,145)
815 PRINT
815 PRINT "WHILE PICKING UP THE" ; " " ; " "
820 CALL HCHAR(23,24,130)
825 PRINT
830 PRINT "PICK UP THE" ; " " ; "TO UNLOCK THE"
835 CALL HCHAR(23,15,140)
840 PRINT
845 PRINT "ORCHARD GATE,"
850 PRINT
855 PRINT "BEWARE-DON'T TOUCH THE" ; " " ; "OR"
860 PRINT
865 PRINT "YOU WILL BE ELECTROCUTED,"
870 PRINT
875 PRINT "WATCH OUT-DON'T LET THE" ; " " ; " "
880 CALL HCHAR(23,27,155)
885 PRINT
890 PRINT "CATCH YOU,"
895 PRINT
895 PRINT "ALSO AVOID THE FARMER'S" ; " " ; " "
900 CALL KEY(0,K,S)
905 IF S=0 THEN 900
910 CALL CLEAR
915 INPUT "TYPE IN YOUR TIME LIMIT--" ; LIMIT
920 CALL CLEAR
925 GOTO 190
```

Hints on conversion

If you are going to try to convert this program to another machine, remember that the Texas has a 24 by 32 screen and it uses user-defined characters. A good amount of space can be saved by using multiple line statements. Here are the more unusual commands:

CALL CHAR (ASCII code number, hexadecimal string). Defines characters used in the program. The hexadecimal string describes the pattern to be given to the character with ASCII code stated

CALL HCHAR (row number, column number, ASCII code, number of repetitions). Places character on screen at row and column specified, and repeats it horizontally the number of times stated. If this fourth value is omitted then the character is displayed only once. The character used is the one whose ASCII code appears in the command

CALL VCHAR (row number, ASCII code, number of repetitions). Works the same way as CALL HCHAR only repeats vertically instead of horizontally

CALL GCHAR (row number, column number, numeric variable). Equivalent of PEEK, assigns ASCII code of character at row and column stated to numerical variable

CALL CLEAR clears the screen. Equivalent to CLS

CALL SCREEN (colour code). Specifies screen colour, using colour code

CALL COLOUR (character set number, foreground colour code, background colour code). Specifies foreground and background colours for all characters in set stated

CALL SOUND (duration, frequency, volume). Produces sound with duration, frequency, and volume specified. Duration is in milliseconds, frequency in Hertz and volume in scale from 0 (loudest) to 30 (quietest)

CALL KEY (key unit, k,s). Equivalent of INKEYS or GET. Returns code if key pressed in variable K.

which unlocks the gate at the top. Make sure the farmer doesn't pick up the key before you.

You score 10 points for every apple picked up, and 50 points if you pick up the key. Your score doubles if you escape through the gate.

You set your own difficulty level, when line 915 asks for your time limit.

You can also vary the game by making the farmer move from bottom to top of the screen. Just change line 380 to FOR R=22 TO 2 STEP -5.

To speed up the game, leave out line 446.

TI-99/4A SOFTWARE REVIEWS

Battle Tanks TI-99/4A £6

Stainless Software, 10 Alstone Road, Stockport, Cheshire SK4 5AH

This is a two-player game for those with a bellicose nature.

You each have three tanks and the aim is to destroy your opponent's hardware.

Dividing you and your opposite number is a series of walls which you can batter down with your shells — up to three walls before your salvoes fall short and you are forced to advance.

Control of your respective tanks is through either the split-keyboard or the joysticks, each being scanned alternately. So it would seem that the player who can keep his/her finger 'on the button' continuously might eventually be the victor.

The choice is not made clear in the embedded instructions, but is provided in the cassette inlay.

The program does remind you to make sure that the troublesome alphalock key is depressed before you commence battle.

The graphics are not dazzling and the instructions are printed slowly, but luckily you don't have to endure them each time round, unlike on some programs.

While it might appeal to children it is too boring for adults, and really needs something to spice it up in order to make it worth its price tag. **P.B.**

instructions	50%
playability	90%
graphics	50%
value for money	40%



Tank Battle TI-99/4A (Extended BASIC plus Joystick) £5.95

Temptation Software, 27 Cinque Ports Street, Rye, Sussex

With five skill levels, described as being from "strategy to desperation", you may well imagine that with Tank Battle you're in for a tough time.

Enemy tanks appear in random formation from the right and top of the screen. Using your tank at the bottom of the screen you must stop five (not 10, as it specifies in the instructions), of the enemy from infiltrating your lines to

Take on your Texas

Your TI-99/4A takes on a variety of hostile forms in these action games, rated for you by our review panel

reach their objective, your H.Q.

You have only three lives with which to do so. Your movements are restricted to east or west, and range of fire is limited to the upper half of the screen, so some fast reactions and judgement are required to keep out the enemy.

For each enemy tank hit, points are awarded, the aim being to attain a high score.

There are also facilities to interrupt play, or changeover joysticks without having to CLEAR

and re-RUN.

Such features show that as much consideration has been given to the overall presentation of the program as in the play itself. **J.W.**

instructions	75%
playability	90%
graphics	90%
value for money	100%



Lunar Base/ Torpedo Fire TI-99/4A £7

Stainless Software, 10 Alstone Road, Stockport, Cheshire SK4 5AH

Leaving the safety of the mothership, you must guide your lunar pod through the asteroid belt to a lunar base standing above a marsh. After landing, your pod must immediately return to its mothership.

On lift-off the base disappears, leaving only marsh. Manoeuvres continue for as long as you have pods, scoring with each successful move.

When all landing bases have disappeared a new, more difficult set up is given and the game progresses.

Unfortunately there is one bug in what would otherwise be a crashproof program — an IN-

CORRECT STATEMENT IN 4340 turned out to be a typing error.

Torpedo Fire sends you down through oceanic depths to a submarine. Here you take charge of a five-position gun with which to torpedo war and cargo ships as they pass across the surface.

There are five classes of vessel, worth different points, and in each game 25 ships. At the end of a run your score and accuracy are displayed.

Left and right movements are controlled by the cursor keys.

Both games have good graphics and sound. Neither is disadvantaged by the inevitable slowness of TI BASIC. **J.W.**

instructions	90%
playability	70%
graphics	80%
value for money	80%



Space Rescue 2.0 TI-99/4A (Extended BASIC) £11

PS Software, from Stainless Software, 10 Alstone Road, Stockport, Cheshire SK4 5AH

The instructions to Space Rescue 2.0 give a full scenario of the game, which is set in the 21st century.

In brief, Earth has been robbed of the space factories and lunar mines which provide its energy. Using the last 10 landers aboard their mothership, you must penetrate the defences of the Xyolians to gather as many power pods as you can.

The pods are placed on three pylons at the bottom of the screen. Each lander is released from the mothership, guided past the crossing alien defences to land

on the pylon, then returned to the ship.

Once all the pods have been collected, the action becomes progressively more difficult as defences are increased.

This lasts through four screens, after which the mothership returns to Earth. The game then restarts at a higher skill level.

Control of the landers is either by joystick or keyboard.

As the lander is moved continuously in one direction it starts to accelerate. To counteract this the direction must be reversed until stopped. This makes manoeuvring more difficult than in a straightforward dodging game. **J.W.**

instructions	100%
playability	80%
graphics	95%
value for money	70%



Galactic Gunfight TI-99/4A (Extended BASIC) £10

By Intersoft, from Stainless Software, 10 Alstone Road, Stockport, Cheshire SK4 5AH

Yes, it's time to take on those aliens again, this time in a

Galactic Gunfight.

Before the game starts, you are given the option of using joystick or keyboard. If the keyboard is selected, you can then decide which of the three keys you want to use for moving the starfighter up and down, and to fire.

Next comes the level of difficulty, which is set from the laser firing time and can be from one to 100 seconds. Twenty is regarded as "Super Pro".

Play then starts, as you attempt

to blast the aliens which appear in squadrons of five.

Since you only have a limited time in which to dispose of your attackers, some strategic play is called for.

Any time remaining when you have destroyed a squadron will be required when the next arrives, particularly as this moves twice as fast.

There is a limit to the number of times your laser can be fired in succession. More than two rapid

bursts will result in overheating for which a cooling period is required, which wastes time.

Quite a good game, made that much better by having such a range of difficulty levels. **J.W.**

instructions	100%
playability	60%
graphics	60%
value for money	60%



You're in charge of the ball and chain

Here's the second part of my Slave Labour game for the unexpanded TI-99/4A. Type it in and find out just how hard a taskmaster you are.

Using the D and S control keys your job is to place a ball and chain in the path of a slave to keep him planting crops in the top half of the screen.

If three slaves escape you're sacked.

Here's part two of Slave Labour, Iain Bartram's novel game for the unexpanded TI-99/4A

How it works

-2020 escaping slaves routine
2030-2070 word centring for screen output
2080-2120 rest, return to main program
2130-2200 slave bouncing off walls and off ball and chain in correct direction
2210-2310 planting crop and determine direction of return
2320-2370 scoring
2380-2450 move to next field
2460-2600 end, play again?

```
1990 NEXT I
2000 RETURN
2010 IF SL=3 THEN 2400
2020 FOR W=19 TO 20
2030 CALL HCHAR(W,1,32,32)
2040 NEXT W
2050 GOTO 970
2060 ADIR=-1
2070 IF E=-2 THEN 2100
2080 BDIR=-1
2090 GOTO 1130
2100 BDIR=1
2110 GOTO 1130
2120 IF (B<2)+(B>31) THEN 2130 ELSE 2140
2130 BDIR=-BDIR
2140 CALL GCHAR(A+ADIR,B+BDIR,BL)
2150 IF BL=144 THEN 2170 ELSE 2160
2160 IF A=6 THEN 2170 ELSE 1060
2170 IF A=10 THEN 1060
2180 ADIR=-ADIR
2190 IF B<2 THEN 2240
2200 IF B>31 THEN 2220
2210 IF RND>.5 THEN 2220 ELSE 2240
2220 BDIR=-1
2230 GOTO 2250
2240 BDIR=1
2250 CALL SOUND(50,-6,1)
2260 SC=SC+A*2-10
2270 K$="SCORE="&STR$(SC)
2280 R=2
2290 GOSUB 1960
2300 IF SC>340 THEN 2310 ELSE 2380
2310 CALL CLEAR
2320 R=8
2330 K$="THIS PLANTATION IS FULL"
2340 GOSUB 1960
2350 R=10
2360 K$="MOVE ALONG TO THE NEXT ONE"
2370 GOSUB 1960
2375 GOTO 960
2380 CALL HCHAR(A,B,144)
2390 GOTO 1140
2400 CALL CLEAR
2410 IF SC>200 THEN 2420 ELSE 2460
2420 R=12
2430 K$="YOUR NOT A BAD TASK MASTER"
2440 GOSUB 1960
2450 GOTO 2490
2460 R=12
2470 K$="YOU CALL YOURSELF A TASK MASTER!"
2480 GOSUB 1960
2481 FOR D=1 TO 50
2482 NEXT D
2490 CALL CLEAR
2500 R=12
```

```
2510 K$="DO YOU WANT TO PLAY AGAIN?"
2520 GOSUB 1960
2530 CALL KEY(0,K,S)
2540 IF K=78 THEN 2600
2550 IF K=89 THEN 2570
2560 GOTO 2530
2570 SC=0
2580 SL=0
2590 GOTO 960
2600 END
```

Micro Tip

Atari

256 colours at once

This interrupt-driven program puts all of the Atari's 256 colours on the screen at the same time.

Here's how it works. As well as showing what graphics mode the next line is, the display list can also give interrupt instructions. (An interrupt is a machine code subroutine called during a vertical or horizontal blank on the screen.)

When this instruction is found, the computer looks at location 54286.

If it contains 192 it finds the routine to be executed in memory by looking in addresses 513 and 512, high and low memory bytes respectively.

Once the interrupt is completed the computer returns to what it was doing before being disturbed.

What my program does is change the colour register in each of the display list interrupts therefore providing 16 colours. The 16 shades of these colours are produced using Graphics 9.

```
10 GRAPHICS 9
20 REM FILL SCREEN
30 FOR X=0 TO 79: C=INT(X/5): COLOR C*(X/5 C)
40 PLOT X,0: DRAWTO X,191: NEXT X
50 REM DISPLAY LIST INTERRUPT MACHINE CODE
60 FOR I=1536 TO 1548: READ D: POKE I,D: NEXT I
70 DATA 72,165,205,141,26,208,24,105,16,133,205,104,64
80 REM PUT INSTRUCTIONS IN DISPLAY LIST
90 DL=PEEK(560)+256*PEEK(561)
100 FOR I=0 TO 15: READ D: POKE DL+D,143: NEXT I
110 DATA 6,16,28,41,53,65,77,89,102,114,126,138,150,162,
174,186
120 REM HORIZONTAL LINES
140 COLOR 0: FOR I=0 TO 191 STEP 12: PLOT
O,Y: DRAWTO 79,Y: NEXT Y
150 POKE 712,0: POKE 512,0: POKE 513,6: POKE 205,0: POKE
54286,192
160 GOTO 160
```

The instruction put in the display list is 128 plus its previous contents. For example 15 + 128, giving 143.

Amin Hoque

Give them a lift... if you can stand the strain

It's a risky job being a lift attendant – in this program anyway. Rod Lane wrote the game for the unexpanded TI-99/4A

My game places you in the role of a despairing lift attendant whose job is to carry passengers in a six-storey office building to the rooftop helipad.

However, your lift is in need of repair and the doors have the unfortunate habit of opening when leaned on — even when the lift is not present — and the passengers take a nasty tumble.

Your task then is to whisk the passengers to the roof as quickly as possible, using the E and X keys.

This is complicated by two factors.

The mechanism is old and weak and if you try to cram more than six people into your lift the cable will snap with obvious consequences.

And when a disaster happens the other workers, realising something is amiss, pour out of their offices into the corridor near the lift shaft.

When more than 10 people have suffered an untimely end the game finishes and displays your tally and the opportunity to try again.

If at any time the strain becomes too much you can break the program by pressing FCTN/CLEAR.

Hints on conversion

TI BASIC is fairly standard with two main exceptions. Single statement lines are used and subroutines for graphics and sound are available, prefixed with CALL, as detailed below.

CALL CLEAR clears screen

CALL SCREEN defines screen colour

CALL CHAR defines a character with a 16-digit hex string from an eight by eight grid. Other machines will use a different method

CALL COLOUR (character set, foreground, background) sets colour of character

CALL HCHAR (row, column, character code, number of repeats) places character on screen at co-ordinates specified and optionally repeats character horizontally

CALL VCHAR similar to HCHAR but repeats character vertically

CALL GCHAR (row, column, variable) similar to a screen PEEK on other machines. Gives the ASCII code at the given screen co-ordinates

CALL SOUND (D,F,V) gives sound of duration D, frequency F and volume V. Three sounds can be produced at once

Variables

F,S,T,FT,FIF,SX column positions of the six figures

COL value of above used in subroutine

MET number of passengers met and therefore in the lift

TOT total of passengers to reach the helipad

LOST number of passengers who failed to make it

DET number of places from the end of the floor on which the figures appear

How it works

140-250 set up variables

260-370 define characters

380-590 draw screen

600-900 scan keyboard, move lift or move passengers and, when lift reaches roof, write total number of passengers

910-1270 randomly decide on which floor the passenger advances and remembers how far along corridor figure has reached. Extra choices included to make progress of the figures less smooth

1280-1380 move figures

1390-1490 check when passenger reaches lift shaft whether lift is present, score success, check for over-loading

1670-1770 crash lift on over-loading

1780-1800 print tally, offer new game



```

100 REM  LIFT ATTENDANT
110 REM  BY R.LANE.
120 REM
130 REM  INITIAL VALUES
140 CALL CLEAR
150 F=31
160 S=31
170 T=31
180 FT=31
190 FIF=31
200 SX=31
210 MET=0
220 LOST=0
230 DET=0
240 TOT=0
250 RANDOMIZE
260 REM  DRAW CHARACTERS
270 CALL CHAR(128,"0808080808080808")
280 CALL CHAR(129,"087F557F557F557F")
290 CALL CHAR(136,"0010301038102868")
300 CALL CHAR(137,"0010301038102C24")
310 CALL CHAR(132,"000000CE142378FF")
320 CALL CHAR(144,"FFFF00000000FF")
330 CALL COLOR(14,6,1)
340 CALL COLOR(15,9,1)
350 CALL COLOR(3,8,1)
360 CALL COLOR(4,8,1)
370 CALL SCREEN(16)
380 REM  SET UP SCREEN
390 CALL VCHAR(1,4,30,24)
400 CALL VCHAR(2,6,30,23)
410 CALL VCHAR(1,5,128,24)
420 P=INT(24*RND)+1
430 CALL HCHAR(P,5,129)
440 FOR ROW=4 TO 24 STEP 4
450 CALL HCHAR(ROW,7,30,25)
460 CALL HCHAR(ROW-1,6,32)

```


TI-99/4A PROGRAM

```

470 NEXT ROW
480 CALL HCHAR(2,7,144,4)
490 M$="MET"
500 C=11
510 GOSUB 560
520 M$="LOST"
530 C=21
540 GOSUB 560
550 GOTO 610
560 FOR L=1 TO LEN(M$)
570 CALL HCHAR(2,C+L,ASC(SEG$(M$,L,1)))
580 NEXT L
590 RETURN
600 REM  Move Lift
610 CALL KEY(3,K,ST)
620 IF ST=0 THEN 920
630 IF K=69 THEN 660
640 IF K=88 THEN 830
650 GOTO 920
660 P=P-1
670 IF P<=0 THEN 720
680 CALL HCHAR(P+1,5,128)
690 CALL HCHAR(P,5,129)
700 CALL SOUND(150,-7,0)
710 GOTO 610
720 P=1
730 TOT=TOT+MET
740 MET=0
750 CALL HCHAR(2,16,48)
760 FOR I=1 TO LEN(STR$(TOT))
770 CALL HCHAR(1,6+I,ASC(SEG$(STR$(TOT),I,1)))
780 NEXT I
790 CALL HCHAR(1,6+I,136)
800 CALL SOUND(150,659,3)
810 CALL SOUND(100,880,0)
820 GOTO 920
830 P=P+1
840 IF P>=25 THEN 890
850 CALL HCHAR(P-1,5,128)
860 CALL HCHAR(P,5,129)
870 CALL SOUND(150,-7,0)
880 GOTO 610
890 P=24
900 GOTO 610
910 REM  Decide On Floor
920 C=INT(10*RND)+1
930 ON C GOTO 940,990,1040,1090,
1140,1190,1240,1250,1260,1270
940 ROW=3
950 COL=F
960 GOSUB 1290
970 F=COL
980 GOTO 610
990 ROW=7
1000 COL=S

```

```

1010 GOSUB 1290
1020 S=COL
1030 GOTO 610
1040 ROW=11
1050 COL=T
1060 GOSUB 1290
1070 T=COL
1080 GOTO 610
1090 ROW=15
1100 COL=FT
1110 GOSUB 1290
1120 FT=COL
1130 GOTO 610
1140 ROW=19
1150 COL=FIF
1160 GOSUB 1290
1170 FIF=COL
1180 GOTO 610
1190 ROW=23
1200 COL=SX
1210 GOSUB 1290
1220 SX=COL
1230 GOTO 610
1240 GOTO 610
1250 GOTO 610
1260 GOTO 610
1270 GOTO 610
1280 REM  Move Passengers Subrou
TIME
1290 CALL HCHAR(ROW,COL+1,32)
1300 IF (COL=5)+(COL-1=5) THEN 14
00
1310 CALL HCHAR(ROW,COL,136)
1320 COL=COL-1
1330 CALL SOUND(60,880,0)
1340 CALL HCHAR(ROW,COL+1,32)
1350 CALL HCHAR(ROW,COL,137)
1360 CALL SOUND(100,587,0)
1370 COL=COL-1
1380 RETURN
1390 REM  Check For Lift
1400 CALL GCHAR(ROW,5,GET)
1410 IF GET<>129 THEN 1510
1420 CALL SOUND(100,440,0)
1430 MET=MET+1
1440 IF MET>6 THEN 1680
1450 FOR I=1 TO LEN(STR$(MET))
1460 CALL HCHAR(2,15+I,ASC(SEG$(STR$(MET),I,1)))
1470 NEXT I
1480 COL=31-DET
1485 IF COL>8 THEN 1490
1486 COL=8
1490 RETURN
1500 REM  Miss Passenger
1510 FOR SOUND=880 TO 220 STEP -
20

```


TI-99/4A PROGRAM

```

1520 CALL SOUND(50,SOUND,0)
1530 NEXT SOUND
1540 DET=DET+3
1550 LOST=LOST+1
1560 FOR I=1 TO LEN(STR$(LOST))
1570 CALL HCHAR(2,26+I,ASC(SEG$(
STR$(LOST),I,1)))
1580 NEXT I
1590 IF LOST>10 THEN 1640
1600 COL=31-DET
1610 IF COL>8 THEN 1630
1620 COL=8
1630 RETURN
1640 FOR DELAY=1 TO 2000
1650 NEXT DELAY
1660 GOTO 1780
1670 REM OVERLOAD SEQUENCE
1680 FOR E=ROW TO 24
1690 CALL HCHAR(E,5,32)
1700 CALL SOUND(50,-6,0)
1710 NEXT E
1720 CALL HCHAR(24,5,132)
1730 CALL SOUND(350,-7,1,110,0,1
15,0,120,0)

```

```

1740 FOR DELAY=1 TO 2000
1750 NEXT DELAY
1760 CALL CLEAR
1770 PRINT "YOU OVERLOADED THE L
IFT."::
1780 PRINT "YOU ELEVATED";TOT;"P
ASSENGERS";"YOU LOST";LOST+MET;"
LIVES.":::
1790 INPUT "PRESS ENTER TO START
AGAIN":A$
1800 GOTO 140

```

Micro Tip

VIC-20

Make space for hi-res

This line can be entered as a direct command, or as part of a program, so as to reconfigure your VIC to take hi-res graphics with a 16K RAM pack.

It can be used in a program of instructions, for instance, to reserve space for hi-res and then to auto-load the next part of the program.

POKE 198,0:POKE 648,30:POKE 642,32:POKE 198,10
:POKE 631,131:SYS 64824
631 is the first location of the keyboard buffer — not 632, as was stated recently in one magazine.

David Shepherdson

ARCADE ACTION - ORIC-1 - ADVENTURES

DINKY KONG

For 48K £6.95

SUPER M/C ARCADE GAME

- Skill Levels, Full Colour, Platforms, Ladders, Fire-balls, Rolling Barrels, Umbrellas, Hearts, Hall of Fame, Sound Effects.

You won't see a better game for the Oric.

*Author — Adrian Sheppard.

JOGGER

For 48K £6.95 inc.

REAL M/C ARCADE ACTION

- 4 Screens, Skill Levels, Full Colour, Road, Cars, Lorries, River, Crocs, Logs, Hall of Fame, Sound Effects.

Perspire your way up the screen.

*Author —
Adrian Sheppard.



Dink Kong available from
selective branches of W.H. Smith.

ORICADE

For 48K £8.50 inc.

- Assembler/ Disassembler/ Editor
- Handles full 6502 mnemonics. Features Save & Reload M/C Assembles & Disassembles at any address

You can't program seriously without Oricade.

*Author — Adrian Sheppard.

GRAIL

For 48K £6.95 inc.

Exciting Graphic Adventure

Where in the Castle Perilous is the Holy Grail? Gather armour and weapons to fight monsters. Sell treasure to a trader in exchange for strength potions and wound ointment. Where will the warp take you to? This is a test of skill, luck, logic & intelligence.

MORIA

For 48K £6.95 inc

A challenging adventure set in the mines of Moria.

- Can you survive encounters with the monsters of Middle-Earth? Will the wizard help you? Are you fated to die beside the sealed doors? Or have you the power to open them? Unless you find Durins Ring you will never leave the mines alive!

*Now available from branches of Laskeys.

*Dealers enquiries welcome — excellent trade terms.

*Ask for Severn Software at your local computer store.

Please send me your.....program.
For my Oric-1

I enclose cheque/ P.O. for £.....

NAME.....

ADDRESS.....

.....Post Code.....

SEVERN SOFTWARE HCW38
5, School Crescent, Lydney, Glos. GL15 5TA.

LETTERS

Logic doesn't cheat anymore!

Following the software review of Logic Systems Draughts in HCW 30 where it was pointed out that the program "cheated" the player, and to which the reviewer concluded "Now that's what I call really human behaviour!", Logic Systems assures HCW that this is not the case with any of the Draughts tapes currently on the market.

An early copy was sent to HCW, which had a fault whereby if you manoeuvred the game so that you could take more than one piece at a go, your piece just disappeared into thin air!

The faulty copies were, of course, replaced, but what with this review and the amused comments we received about this fault, I'm not so sure that we shouldn't have offered it as a feature! Logic Systems, Cherry Hinton, Cambridge

Is this the end?

The announcement that Texas Instruments is to cease production of the TI-99/4A must have alarmed many TI owners.

Owners may be assured that TI has its guarantee commitments in mind, and will continue to provide service for at least the next year.

From TI we learn: "Texas Instruments will of course carry on supporting the TI-99/4A in terms of service/repair and there is plenty of software and accessories to satisfy existing owners and those who purchase up to Christmas..."

Present owners should remember that no new TI modules or peripherals will be produced and buy what they need within the next month. Supplies of some items are very low.

We understand that there are many surplus consoles and these are likely to be reduced in price, as are some of the games modules, to clear stock before the end of the year.

The more costly modules and the peripherals are not in such surplus, however. If you are thinking of buying these you should obtain a loan and buy them while you can!

Galaxy Video in Maidstone have very small supplies of a very recently released module LOGO2, and some new Funware modules (Ambulance and Driving Demon). Owners

Send your letter to Letters, Home Computing Weekly, 145 Charing Cross Road, London WC2H 0EE. Don't forget to name your computer — the best letter could win £5-worth of software. Got a problem with your micro? We'll soon be starting a queries page, so send your technical questions to Queries at the above address



should support such dealers!

If owners wish to maintain support, it is suggested that all requests be channelled through the three or four dealers who have shown practical support for the computer.

By concentrating demand, we can make it worth their while to consider the import of third party modules and peripherals from the States... always a risk business.

Here in the UK we have a third party supplier of 32K RAMs (Arcade Hardware) and they, together with others, also supply joysticks or adaptors. Galaxy lists over 20 titles of books, and also supplies 99er Magazine.

Stainless Software will continue to supply cassette programs so long as there is demand. We understand the other principal software houses will be continuing. Stainless Software still welcomes additions to its catalogue, by individual programmers or from small suppliers who now wish to leave the market.

But all enquiries to us from individuals must be accompanied by an SAE! Stephen Shaw, Stainless Software, 10 Alstone Road, Stockport, Cheshire SK4 5AH

The ultimate frustration?

Some of the software houses charge more for their games and perhaps include booklets etc. in their package.

Ultimate, who have produced some excellent games this year, should have charged more for their current game Jetman and included a bottle of Scotch and a swear box!

If I ever get to pick up that damn bomb I shall know exactly where to drop it — The Green, Ashby de la Zouch.

Thanks, anyway, to Ultimate for top quality games.

Roy Bristow, Bradford-on-Avon, Wilts

Long wait for Micronet

Please warn any readers hoping to join Micronet in the future that they may be in for a very long wait, as it has taken me just over two months to get in!

I first of all filled in the form which came with my Prism modem, and posted it off first class. I waited for three weeks and heard nothing, and so decided to give them a ring and find out what was happening. I was told that it would take between three and four weeks to deal with my application, which I accepted.

But another two weeks went by and still nothing had happened, so I gave them another call. They told me I would get my number within the next 10 days, as they were very busy (the advert says the whole process takes 28 days).

About five days later, I received a call asking me for another £100 for my modem. I explained that I already had a modem and just wanted my number, and the girl I spoke to said that she would deal with it.

She did that alright, as the next time I rang up, they had lost my form. After this I thought things could only get better. How wrong could I be?

When they eventually found my form, I was promised that the number would be sent that day, first class. My number finally arrived two months later, and gratefully received.

I immediately went and entered my number into the computer only to find that it was not my number at all, and so would not work. The right number arrived eventually — but without instructions or directory, which took another week to come.

Vicki Blundell, Potters Bar, Herts

A Micronet spokesman said: "The service has been bad, there's no denying it, but we are trying our best to improve it — and practice makes perfect. You can currently expect to wait about six weeks, but we're hoping to reduce that soon. One problem has been that the application form is fairly complex, and people have filled it in wrongly, which has slowed the process down. Secondly, bear in mind that the time it takes to connect up to Micronet is not entirely under our control — it partly depends on British Telecom, who have to come and fit the jack plugs."

Scoring high...

I was pleased to see you are reviewing modules for the TI-99/4A and carrying a regular Texas program.

Since many Texas owners are writing in to tell you about their high scores on Parsec, mine is 201,000. But what I'd really like to know is whether anyone has beaten my high score on the Munchman module: 212,500. Munchman gets very fast, especially when you get on to the 60th sheet. **Lee Lambert, Hull, Humberside**

...with cartridges

I read in HCW 35 that Paul Hopkins, a TI-99/4A owner, had achieved a score of 1,080,500 on Parsec, and wanted to know if this was an all-time high.

In fact the highest score I know of was by Cody Collins of Arlington, Texas, who wrote into the 99er magazine Hall of Fame. My own record on Parsec is 702,100.

My favourite TI cartridges are Alpiner (reviewed in HCW 30) and TI Invader. I achieved a score of 115,959 on the third level of Alpiner — this level is quite ridiculous, as you can't move without an avalanche hitting you. On TI Invader, my record is 16,150.

And I'd also like to know whether anyone can beat these scores: Car Wars — 28,520 and Munchman — 151,680.

If you're thinking of buying a joystick for your TI-99/4A, don't buy the TI joystick — it's very uncomfortable and has slow reactions. I have bought the Suncom TAC2 joystick and adaptor, which I think is much better.

Julian Shawcross, Boothstown, Manchester

A series of disasters

I am having difficulties with Issue 3 Spectrums which have no connection with the published IN function problem.

So far, I have had three Issue 3 48K Spectrums, all of which seem to have an intermittent fault, which seems unrelated to overheating, since it can occur within minutes of power-up, or to memory limit, since it can occur on quite small programs. When present, the following happens:

1. Moving graphics, which have previously run without difficulty, now fail to print the leading or trailing space. Over-printed graphics fail to erase those previously printed. It is as if the OVER function has been switched on.
2. Out of Memory appears when trying to re-load an under-2K BASIC program, only just Saved and Verified.
3. The Load Search Border sequence can suddenly start in the middle of running a professional program.
4. The C cursor changes to an asterisk and will only print rubbish. If delete is used, the screen scrolls one character space left, the lost first letters re-entering on the right, and then crashes. This is an adventure program.
5. The K cursor, when listing, disappears by itself and coloured squares appear in the listing, with an Invalid Colour message.
6. When listing, a 'black window' appears in the middle third of the screen, displaying flashing coloured squares and random letters.
7. Entering LIST produces an Invalid Stream message (I don't have a Microdrive) All efforts to get back into the listing or run the program stops with more Invalid Stream messages.
8. Programs, written in BASIC, on the same computer, which have run without problems many times, stop with Nonsense in BASIC messages, referring to perfectly legitimate lines.
9. Listing will not accept NO to scroll
10. Graphic figures are unintentionally erased.
11. The program NEWS by itself.

Seven of the above problems have been common to all three Issue 3s, which makes me wonder if the new chip has side-effects unsuspected by Sinclair.

Are any of your readers having similar difficulties with Issue 3s?

Dorene Cox, Dagenham, Essex

Fun for all the family

I own a TI-99/4A and like Patricia Bates (HCW 31), I am addicted to my computer. The children love it, and I spend more time keeping up with them than on my own games.

But my husband was totally disinterested in

computing — until we bought a Video Chess cartridge. "It does do something after all", he says, after beating the TI-99/4A at chess first game.

I too like Patricia Bates, am plodding away, learning programming — but in a couple of years I'll be able to compile bits and pieces. I am pretty much self-taught, with the help of the excellent TI Users Book and the TI Users Group — grateful thanks to Terry Grimshaw.

I cannot praise the TI people at Bedford enough. I had problems with my modulator, but didn't want to part from it. Since it was under guarantee, I was promptly sent another one, and loads of literature.

I'd like to see more women sending in programs — and not just games. For example, what about some simple programs for pre-school children — or a how-to-murder-your-husband computer program?

A. Bates, Havant, Hants

Boxing clever

As someone new to home computing, I have recently bought examples of all the usually available magazines.

The support that HCW gives to the TI-99/4A as illustrated by "Save us from an Alien Invasion" and the profile of Stainless Software (issue 27) makes it a firm favourite of mine destined for regular reading.

One disadvantage I have noticed with all programs listed in magazines is that once they're on tape and working, only a rough home-produced label is available for future reference on the cassette box.

May I suggest that all listings you produce are accompanied by a cassette box sized printed picture and title with instructions for use on the reverse?

I am sure other users would find this a useful aid to remembering what each program is. It also gives a record of program source — good future advertising for you and the programmers!

R Brunt, Gloucester

It's a good idea, but unfortunately as space is at a premium in HCW, we wouldn't have room to include an insert specially for each listing. However, we do plan to print more of the colourful inserts that appeared in HCW 29, for readers to cut out and use with their own programs. And those who own a VIC-20 and a printer could always print their own, using Alan Blackham's program in HCW 30.

Is TI its own worst enemy?

Like most TI-99/4A owners I have spent, and wasted a lot of time trying to find good quality independent software for my machine and I am beginning to think that the problem may be self-perpetuating.

If little software is available, perhaps TI owners are learning to program for themselves and discovering that this is infinitely more satisfying than buying and playing even the best games available.

This may reduce the market to the point where the big software houses won't consider producing TI software, leaving supply in the hands of the small independent producers, some of whom have marketed some first-class products.

To move on to TI's own marketing, I recently visited three local chain stores, all advertising the TI-99/4A as being in stock. The first had the machine but no Texas software at all, and still has none six weeks later.

The second had an excellent selection, all priced about 25 per cent higher than I have ever seen before. The third shop had a few forlorn modules on the bottom shelf and an assistant who, when approached said: "Oh, have you got a Texas? Perhaps you can help me. I've just bought one and I need some advice..."

Surely my experience is not typical... Is it?

Graham Baldwin, South Croydon, Surrey

Encourage me to buy

High street computer dealers will have to give themselves a hard kick up the backside if they are to clear their shelves.

As a complete newcomer to the scene, I have visited various shops for advice on what I should buy and what the apparatus will do.

Not a single one has offered to give me a demonstration, although I am obviously a sincere prospective buyer, and my questions have met with horribly vague answers such as: "Well, you know like, it's difficult to say. I suppose they'll do most things, so to speak. I'm not quite sure, really."

One shop had an Atari with a permanent display which announced $13 + 7 = 20$. Another was equipped with a Spectrum attached to a TV showing an amateurish drawing of the planet Saturn and a printout which said

"Input error". The following day it depicted nothing except a non-stop snowstorm of black and white stripes. Day 3: switched off.

What encouragement is there to spend the £150-180, which was what I had in mind?

So-called beginners' books are no better. They all start off by announcing they are for the average man in the street — but after a few pages assume the reader is managing director of a multinational firm with 3,286 customers in Outer Mongolia alone.

There must be many other people who have decided to delay buying until things get put on a more business-like basis.

G. H. Willett, Basildon, Essex

Treasure island disc

I've got a great idea for Lance Booth's Treasure Island game (HCW30). When you are eaten by a shark, or drowned in the river, or lost in the mountains, a Death March plays.

Just add the following program lines, making sure the line numbers are correct:

```
156 DATA 294,400,294,
300,294,240,294,300,
349,400,330,300
157 DATA 330,350,294,
350,294,300,277,350,
294,500,0,0
```

```
1331 GOSUB 2000
1361 GOSUB 2000
1391 GOSUB 2000
1421 GOSUB 2000
1451 GOSUB 2000
1481 GOSUB 2000
2000 RESTORE 156
2010 READ NOTE,DUR
2020 IF(NOTE=0)*(DUR=0) THEN 2050
2030 CALL SOUND(DUR,NOTE,1)
2040 GOTO 2010
2050 RETURN
John Macdonald
```

Don't abandon the Atari

All right, own up. You've abandoned the Atari, haven't you? Your magazine is going all Spectrum and BBC like all the others.

Although I realise it's up to the reader to send in programs, you have hardly printed a review either?

A couple of days ago I received a list from Atari of books and magazines that cater for their machine. There were only two English magazines on the list — and HCW was one of them. So come on, buck your ideas up. **Steven Hurst, Bolton, Lancs** Just for you, there's an Atari program in this week's issue

ATARI PROGRAMS

```

724 IF Z<N56 THEN Z=N56
726 IF Y<N34 THEN Y=N34
728 IF Y>N130 AND GM=N1 THEN Z=96
730 IF GM=N2 THEN RETURN
734 IF Y>162 AND GM=N3 THEN Y=162
740 A$(Y,Y+35)=B$:POKE 53248,Z:RETURN
1000 SOUND N0,255,N10,8:FOR I=N1 TO N52:
NEXT I:SOUND N0,N0,N0,N0:RETURN
1100 FOR I=N1 TO N2:SOUND N0,20*(PICK+
N1),N10,N4:SOUND N1,20*(PICK+N2),N10,N
4
1110 FOR H=N1 TO 8:NEXT H:NEXT I:SOUND N
0,N0,N0,N0:SOUND N1,N0,N0,N0:RETURN
1200 U=PEEK(709):FOR I=N0 TO N3:FOR H=N3
TO N10 STEP 0.5:ON GM GOTO 1210,1212,12
14,1216,1218,1220
1210 SOUND I,200/(I+GM),N12,N12-H:NEXT H
:NEXT I:GOTO 1240
1212 SOUND I,240/(I+GM),N12,N12-H:NEXT H
:NEXT I:GOTO 1240
1214 SOUND I,N4*(I+GM),N2,N12-H:NEXT H:N
EXT I:GOTO 1240
1216 SOUND I,120*(I+GM),N10,N12-H:NEXT H
:NEXT I:GOTO 1240
1218 SOUND I,120/(I+GM),N12,N12-H:NEXT H
:NEXT I:GOTO 1240
1220 SOUND I,220/(I+GM),N10,N12-H:NEXT H
:NEXT I:GOTO 1240
1240 FOR I=N0 TO N3:POKE 709,16-I*I:FOR
H=N1 TO N10:SOUND I,(N10+I*GM),N12,N10-(
H*(GM+N6)):FOR A=N1 TO N3:NEXT A
1250 NEXT H:NEXT I:POKE 709,U:FOR I=N0 T
O N3:SOUND I,N0,N0,N0:NEXT I
1300 SOUND N0,N10,N0,N4:FOR H=N1 TO N5:N
EXT H:SOUND N0,N0,N0,N0:RETURN
8000 N1=1:N0=N1-N1:N2=N1+N1:N3=N2+N1:N4=
N3+N1:N5=N4+N1:N6=N5+N1:N7=N6+N1:N10=N5*
N2:N11=N10+N1:N12=N10+N2
8005 N13=N12+N1:N14=N12+N2:N17=N12+N5:N2
2=N17+N5:N23=N22+N1:N34=N17*N2:N52=52*N5
6=56:N130=130
8010 PS=N0:DIM P(N52),SU$(N1),NU$(N2),D(
N7,N6),CARD$(40),BLANK$(40),FD$(40),S(25
):OPEN #N1,N4,N0,"K":PICK=N1
8020 CARD$="-----|-----|-----|
|-----|-----|-----|
|-----|-----|-----|
|-----|-----|-----|
8030 FD$="-----|-----|-----|
|-----|-----|-----|
|-----|-----|-----|
|-----|-----|-----|
8040 RETURN
8050 A$(1)="♥":A$(512)="♥":A$(2)=A$:GOSU
B 740:RETURN
8060 FOR I=N0 TO N7:FOR ROH=N0 TO N6:D(I
,ROH)=N0:NEXT ROH:NEXT I:RETURN
8100 GRAPHICS N1:POKE 711,12:POKE 559,62
:POKE 752,N1:POKE 710,116:POKE 712,116:P
OKE 53248,N0
8101 ? #N6:"-----":? #N6
8102 POKE 764,255:GM=N0:? #N6:" SELEC
T GAME"
8103 ? #N6:? #N6:" r HOLE IN ONE":? #N6
:" - PACKET BOAT":? #N6:" + BLOCK ELEV
EN":? #N6:" ● REVERSE PUZZLE"
8104 ? #N6:" ● MONTE CARLO":? #6:" I P
OKER SOLITAIRE":? #6
8105 ? #N6:" during play press option
-to restart game":? #N6:" star
t -to return to menu"
8109 GET #N1,GM:L=GM+256:GM=GM-48:IF GM<
N1 OR GM>N6 THEN 8109
8110 POKE 704,N0:GOSUB 1100:POSITION N2,
N0:? #N6:"selected game ":CHR$(L):GOSUB
9000:GRAPHICS N0:POKE 752,N1
8120 ? "":GOSUB 400:POKE 559,62:POKE 75
2,N1:ON GM GOTO 600,2000,3000,4000,5000,,
6000
8200 GRAPHICS N0:POKE 82,N0:DL=PEEK(560)
+256*PEEK(561)+N4:POKE 752,N1:POKE 710,3
4:POKE 709,0:POKE 712,N0
8210 POKE 708,200:FOR I=N2 TO N11:POKE D
L+I,N5:NEXT I:FOR I=N12 TO 17:POKE DL+I,
N6:NEXT I
8226 POSITION N0,N10:?"XXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXX":POSITION N3,N
12:?"BY r.l.howarth"
8230 FOR I=N0 TO 35 STEP N5:POSITION I,N
4:?"CARD$:NEXT I:FOR I=N1 TO 36 STEP N5
8240 READ SU$:SOUND 0,ASC(SU$)-I,N10,N4:
SOUND N1,ASC(SU$)-I+N1,N10,N4:POSITION I
,N5:?"SU$:FOR H=N1 TO N7:NEXT H
8250 POSITION I+N1,N7:?"♦":NEXT I:SOUND
N0,N0,N0,N0:SOUND N1,N0,N0,N0:FOR I=N1
TO 70:NEXT I:POKE 82,N2
8260 RESTORE 8310:FOR I=N0 TO N10:READ C
:POKE 1536+I,C:NEXT I:POKE 512,N0:POKE 5
13,N6:RETURN
8300 DATA P,A,T,I,E,N,C,E
8310 DATA 72,169,22,141,10,212,141,24,20
8,104,64
9000 ? " DO YOU REQUIRE INSTRUCTIONS? Y/
N":?
9010 POKE 764,255:GET #N1,H:IF H=78 THEN
10020
9020 IF H<89 THEN 9010
9030 GRAPHICS N0:POKE 710,N4:POKE 559,62
:POKE 752,N1:ON GM GOTO 9100,9200,9300,9
400,9500,9600

```

```

9200 ? " 28 cards are dealt in seven col
umns, only the bottom card of each colum
n is displayable ";
9210 ? "The object of the game is to col
lect packets of 4 cards of the same deno
mination.Cards can be played";
9220 ? " on each other if they are simi
lar. When 4 similar cards have been bro
ught together,";
9230 ? " they are removed by placing th
e cursor on the bottom card and press
ing the button. Cards are dealt";
9240 ? " from the pack by placing curs
or on pack and pressing button. A spa
ce can only be filled by a card";
9250 ? " from the discard pile. Pack
is redealt from discards":GOTO 10000
9300 ? " 9 Cards are dealt in a square.
The object of the game is to get rid
of all the pack,";
9310 ? "by taking any 2 cards which add
to eleven. Picture cards can only be
taken when Jack,Queen,King ";
9320 ? "are on the board at the same tim
e. Spaces can be filled from the pack
by placing cursor on a space ";
9330 ? "and pressing button.":GOTO 10000
9400 ? " 20 Cards are dealt face up in a
large square"
9410 ? " The object of the game is to en
d up with the cards in the layout all f
ace down,";
9420 ? " Place the cursor over any card
and press button to turn face down.":?
" Count this as 1."
9440 ? " Now move in a clockwise directi
on counting 2,3,4. Turn the fourth ca
rd face down. Choose ";
9450 ? "any other card,count 1 and turn
down.Turn down the fourth card and carry
on turning 1 and 4 "
9460 ? "until all the cards are reversed
":GOTO 10000
9500 ? "25 Cards are dealt in a 5x5 layo
ut.The object of the game is to pair off
all the cards.A pair can only ";
9510 ? "be taken,if they are next to ea
ch other in any direction-":? " VERT
ICAL,HORIZONTAL or DIAGONAL."
9520 ? "Place cursor on card and press b
utton. If ";
9530 ? "you cant go,then place cursor on
pack and press button.The cards in the
layout will be re-dealt ";
9540 ? "minus any spaces.The layout wi
ll be increased to 25 cards from the dec
k":GOTO 10000
9600 ? "25 Cards are dealt from the stoc
k one by one.Place each card anywhere on
a 5x5 grid to make the best ";
9610 ? "poker hands.Once laid,a card can
not be moved.":? "When the grid is fille
d,the score willbe calculated ";
9620 ? "for 10 hands-":? " 5 across,5 d
own and 2 diagonal.":? "The object being
to get the best score possible"
10000 ? :? "PRESS START TO BEGIN"
10010 IF PEEK(53279)>N6 THEN 10010
10020 GOSUB 1100:?"":RETURN

```

Listing 2 — Poker Patience

```

5990 REM Game 6
6000 POKE 710,64:POKE 712,70:T=N1:GU=32:
POKE DL+N17,N2:POKE DL+N22,130:POKE 5428
6,192:POKE 704,146
6015 FOR I=N1 TO 24:S(I)=N0:NEXT I:POKE
82,26
6020 POSITION 26,0:?" ROY'FLUSH=60
STR'FLUSH=50 F
OURS =30 "
6025 ? " STRAIGHT =20
FULLHOUSE=16 F
LUSH =12 "
6030 ? " THREES = 8 THO

```

```

PAIRS = 5 ONE PAIR = 2 ";
6040 POKE 82,N2:POSITION N10,N23:?"POKE
R SOLITAIRE":POSITION 27,N23:?"HI SCOR
E=";HI:FOR ROH=N2 TO 18 STEP N4
6050 FOR I=N3 TO N23 STEP N5:POSITION I,
ROH:?" ":GOSUB 1300:NEXT I:NEXT ROH:T=N
1:GOSUB 8060
6080 POSITION 31,N17:?"CARD$:C=P(N1):GOS
UB 500:POSITION 32,18:GOSUB 200:GOSUB 13
00
6085 Y=N34:Z=N56:GOSUB 8050
6090 GOSUB 700
6092 IF Z>136 THEN Z=136
6098 GOSUB 724:GOTO 6090
6100 A=INT(Z/20)-N1:R=INT(Y/32)-N1:IF D(
A,R)<N0 THEN GOSUB 1000:GOTO 6090
6110 D(A,R)=P(T):A1=A*N5-N3:R1=R*N4:POSI
TION A1,R1:?"CARD$:IF Y<162 AND D(A,R+N
1)<N0 THEN ? "****"
6120 C=P(T):GOSUB 500:POSITION A1+N1,R1+
N1:GOSUB 200:GOSUB 1300
6130 T=T+N1:IF T=26 THEN 6200
6140 C=P(T):GOSUB 500:POSITION 32,18:GOS
UB 200
6160 GOTO 6090
6200 POP:POSITION 31,N17:?"FD$:PS=N0:PO
SITION N2,23:?"
":FOR U=N1 TO N4
6210 FOR ROH=N0 TO N4:T5=N1:T2=N0:T1=N0:
C1=N0:R5=N0:D1=N0:FOR I=N1 TO N13:PC(I)=N
0:S(I)=N0
6220 NEXT I:POSITION N2,N23:?"
":IF U=N4 THEN FOR I=N5 TO N1 STEP
-N1:C=D(I,ROH):GOTO 6230
6221 FOR I=N1 TO N5:IF U<N2 THEN C=D(I,
ROH):GOTO 6230
6222 IF U=N2 THEN C=D(ROH+N1,I-N1)
6230 GOSUB 500:S(I)=C:P(C)=P(C)+N1:IF I=
N1 THEN R5=SUIT
6235 IF U>N2 THEN ROH=ROH+N1
6240 D1=D1+(SUIT=R5):NEXT I:IF D1=N5 THE
N PS=PS+N12:C1=N2:GOTO 6340
6250 FOR I=N1 TO N13:IF P(I)>N1 THEN T2=
T2+(P(I)=N2)+N10*(P(I)=N3)+100*(P(I)=N4)
6255 NEXT I:T1=N4+(T2=N11)-(T2<N11)-(T2<
N10)-(T2<N2)-(T2<N1):POSITION N2,23
6330 PS=PS+N2*(T1=N1)+N5*(T1=N2)+8*(T1=N
3)+16*(T1=N5)+30*(T1=N4):IF T1<N0 THEN
6360
6340 POSITION N2,N23:S1=S(I):LO=S1:FOR H
=N2 TO N5:IF S(H)>S1 THEN S1=S(H)
6343 IF S(H)<LO THEN LO=S(H):T5=H
6345 IF ABS(S1-LO)>N4 THEN 6352
6348 NEXT H:C1=C1+N1:IF C1=N3 AND LO=N10
AND S1=N14 THEN ? "ROYAL":PS=PS+N10:GO
TO 6350
6349 ? "STRAIGHT";
6350 PS=PS+20:IF C1=N3 THEN ? " FLUSH":
PS=PS+18
6351 GOTO 6355
6352 IF LO=N1 AND S1>9 THEN S(T5)=N14:GO
TO 6340
6353 IF C1=N0 THEN ? "NO SCORE";
6355 IF C1=N2 THEN ? "FLUSH";
6360 IF T1=N1 THEN ? "1 PAIR";
6361 IF T1=N2 THEN ? "2 PAIRS";
6362 IF T1=N3 THEN ? "3 OF A KIND";
6363 IF T1=N4 THEN ? "4 OF A KIND";
6365 IF T1=N5 THEN ? "FULL HOUSE";
6370 POSITION 16,N23:?" SCORE=";PS:NEX
T ROH:NEXT U
6372 FOR I=N1 TO N10:NEXT I:POSITION N2,
N23:?"
6374 FOR I=N1 TO N6:POSITION N17,N23:?"
SCORE=";FOR H=N1 TO N10:NEXT H:POSITION
N17,N23:?"SCORE=";NEXT I
6380 IF PS>HI THEN HI=PS:GOTO 6400
6400 IF PEEK(53279)=N6 THEN ? "":GOTO 8
100
6410 IF PEEK(53279)=N3 THEN ? "":GOTO 8
120
6420 GOTO 6400
6480 FOR I=N1 TO N6:POSITION 27,N23:?"H
I SCORE=";HI:FOR H=N1 TO N6:NEXT H:POSI
TION 27,N23:?"HI SCORE=";
6490 NEXT I:GOSUB 1200:GOTO 6400

```

Micro Tip

Sound of music

I have written a musical sound program for the TI-99/4A. Any key with an ASCII code pressed will play a musical note.

```

10 CALL KEY (0,K,S)
20 IF S=0 THEN 10
30 K=K*5
40 CALL SOUND (100,K,0)
50 GOTO 10

```

Line 30 can be changed to K=K*25 or any other number. The lower the number, the lower the frequency.

Sarah Finch

There's something evil deep in the cavern...

Cavern Attacker needs skill, concentration, judgement... and a little luck.

A sinister race of aliens is attacking your planet and your mission is to destroy the enemy nerve centre sited in the heart of the alien planet.

But the nerve centre can move and tries to trace every move you make. To outpace it you must either hover or accelerate. But beware — this uses a lot of fuel.

With the best fighters engaged in conflict with your planet the nerve centre is guarded by just the cavern and one inexperienced alien.

You have to score a direct hit to destroy the nerve centre. There's a big bonus for hitting the nerve and another for your remaining fuel.

You lose one unit of fuel every time you enter the cavern and two units each time you hover or accelerate.

Each time you leave the cavern you move down a level. Your laser will destroy sections of the cavern and the alien.

When you hit the alien — which can only move up or down — another one appears. This alien can sometimes destroy layers of the cavern and aid your attack.

How it works

20-50 skill levels
60-170 set variables
180-270 define characters, colour
280-570 set up screen
580-690 display, move nerve centre
700-830 display, move bombs
840-990 read keyboard
1000-1040 if keys pressed GOTO
1050-1260 movement of astro-fighter
1270-1500 alien ship, laser
1510-1800 acceleration
1810-1960 astro-fighter's laser
1970-2010 astro-fighter's fuel limit
2020-2370 endings
2380-2420 check to see if alien is hit
2430-2930 instructions

Variables

skill, fuel, score, bonus explain themselves
time number of moves
lza, a, g, c FOR...NEXT loops
b, laz, rrr random numbers
ship, dec, hold, z, zb, zs decision variables
xs x co-ordinate of alien
ye y co-ordinate of nerve centre
x,y; xb, yb co-ordinates of astro-fighter; nerve centre
sec length of laser
k read keyboard

Our planet is under threat again. You can save us by destroying the alien nerve centre, sunk deep under the surface of the enemy planet.

Gary Jowett and David Trillwood wrote Cavern Attacker for the standard TI-99/4A

```
10 GOSUB 2430
20 PRINT "INPUT SKILL LEVEL(1-6)1-EASY"
30 INPUT SKILL
40 IF SKILL>6 THEN 20
50 IF SKILL<1 THEN 20
60 MSG$=" YOU HAVE CRASHED YOUR SHIP MISSION FAILED "
70 FUEL=32
80 SCORE=0
90 REM G, J & D, T
100 REM SET UP SCREEN
110 RANDOMIZE
120 CALL CLEAR
130 TIME=0
140 XS=11
150 BONUS=1500
160 CALL SCREEN(0)
170 SHIP=0
180 CALL COLOR(2,11,1)
190 CALL CHAR(45,"000000AA55000000")
200 CALL CHAR(120,"FFFFFFFFFFFFFFFF")
210 CALL COLOR(11,14,1)
220 CALL CHAR(115,"FFFFFFFFFFFFFFFF")
230 CALL CHAR(121,"FFFFFFFFF7E3C10")
240 CALL CHAR(122,"183C7EFFFFFFFF")
250 CALL COLOR(12,15,1)
260 CALL COLOR(5,7,1)
270 CALL CHAR(64,"0000E213FF13E200")
280 FOR A=3 TO 31
290 B=INT(RND*8)+12-SKILL
300 FOR C=24 TO B STEP -1
310 CALL HCHAR(C,A,120)
320 NEXT C
330 CALL HCHAR(B,A,122)
340 NEXT A
350 FOR A=3 TO 31
360 B=INT(RND*8)+2
370 FOR C=1 TO B
380 CALL HCHAR(C,A,120)
390 NEXT C
400 CALL HCHAR(B,A,121)
410 NEXT A
420 CALL HCHAR(1,3,70)
430 CALL HCHAR(1,4,85)
440 CALL HCHAR(1,5,69)
450 CALL HCHAR(1,6,76)
460 CALL CHAR(35,"0000107EFF000000")
470 FOR A=7 TO 32
480 CALL HCHAR(1,A,115)
490 NEXT A
500 CALL COLOR(1,9,1)
510 CALL COLOR(9,3,1)
520 CALL CHAR(97,"00C0FEFF00000000")
530 CALL COLOR(10,2,1)
540 CALL CHAR(104,"0000543838381000")
550 YE=1
560 X=11
570 Y=1
580 REM MAIN PROGRAM
590 REM
600 REM
610 TIME=TIME+1
620 CALL HCHAR(23,YE,32)
630 SCORE=SCORE+1
640 RANDOMIZE
650 BONUS=BONUS-1
660 IF YE<32 THEN 680
670 YE=1
680 YE=YE+1
690 CALL HCHAR(23,YE,35)
700 IF DEC=0 THEN 830
710 CALL HCHAR(XB,YB,32)
720 XB=XB+1
730 IF XB<24 THEN 760
740 DEC=0
750 GOTO 790
760 CALL GCHAR(XB,YB,ZB)
770 IF ZB=32 THEN 820
780 IF ZB=35 THEN 2160
790 CALL HCHAR(XB,YB,32)
800 DEC=0
810 GOTO 830
820 CALL HCHAR(XB,YB,104)
```

```
830 CALL HCHAR(X,Y,32)
840 CALL KEY(K,S)
850 IF K=81 THEN 1000
860 IF K=76 THEN 880
870 GOTO 890
880 GOSUB 1010
890 IF K=90 THEN 1020
900 IF DEC=1 THEN 950
910 IF K=32 THEN 1040
920 IF K=61 THEN 940
930 GOTO 950
940 GOSUB 1610
950 IF K=80 THEN 970
960 GOTO 980
970 HOLD=1
980 GOTO 1090
990 GOTO 1100
1000 X=X-1
```

Hints on conversion

If you are going to try to convert this program to another machine, remember that the Texas has a 24 by 32 screen and it uses user-defined characters. Here are the more unusual commands:

CALL CHAR (ASCII code number, hexadecimal string). Defines characters used in the program. The hexadecimal string describes the pattern to be given to the character with ASCII code stated

CALL HCHAR (row number, column number, ASCII code, number of repetitions). Places character on screen at row and column specified, and repeats it horizontally the number of times stated. If this fourth value is omitted then the character is displayed only once. The character used is the one whose ASCII code appears in the command

CALL VCHAR (row number, ASCII code, number of repetitions). Works the same way as **CALL HCHAR** only repeats vertically instead of horizontally

CALL GCHAR (row number, column number, numeric variable). Equivalent of **PEEK**, assigns ASCII code of character at row and column stated to numerical variable

CALL CLEAR clears the screen. Equivalent to **CLS**

CALL SCREEN (colour code). Specifies screen colour, using colour code

CALL COLOUR (character set number, foreground colour code, background colour code). Specifies foreground and background colours for all characters in set stated

CALL SOUND (duration, frequency, volume). Produces sound and duration, frequency, and volume specified. Duration is in milliseconds, frequency in Hertz and volume in scale from 0 (loudest) to 30 (quietest)

CALL KEY (key unit, k,s). Equivalent to **INKEY\$** or **GET**. Returns code if key pressed in variable K.

```
1010 GOTO 1100
1020 X=X+1
1030 GOTO 1100
1040 CALL GCHAR(X,Y,Z)
1050 XB=X
1060 YB=Y
1070 IF Z=35 THEN 2160
1080 DEC=1
1090 IF HOLD=1 THEN 1110
1100 Y=Y+1
1110 CALL GCHAR(X,Y,Z)
1120 IF Z<32 THEN 1520
1130 IF HOLD<>1 THEN 1150
1140 FUEL=FUEL-1
1150 HOLD=0
1160 IF X>21 THEN 1200
1170 IF X>3 THEN 1210
1180 X=3
```


TI-99/4A PROGRAM

```

1190 GOTO 1210
1200 X=21
1210 IF Y<32 THEN 1260
1220 FUEL=FUEL-1
1230 Y=Y+1
1240 GOTO 1970
1250 X=X+1
1260 CALL HCHAR(X,Y,97)
1270 CALL HCHAR(XS,32,64)
1280 RRR=INT(RND*100)+1
1290 IF RRR<90 THEN 580
1300 LAZ=INT(RND*20+12)
1310 IF LAZ>20 THEN 1400
1320 FOR LZA=31 TO 32-LAZ STEP -1
1330 CALL HCHAR(XS,LZA,45)
1340 CALL GCHAR(XS,LZA-1,ZS)
1350 IF ZS=97 THEN 2100
1360 NEXT LZA
1370 FOR LZA=32 TO 32-LAZ STEP -1
1380 CALL HCHAR(XS,LZA,32)
1390 NEXT LZA
1400 CALL HCHAR(XS,32,32)
1410 IF LAZ>25 THEN 1430
1420 GOTO 1470
1430 XS=XS-1
1440 IF XS>2 THEN 1480
1450 XS=3
1460 GOTO 1500
1470 XS=XS+1
1480 IF XS<23 THEN 1500
1490 XS=22
1500 CALL HCHAR(XS,32,64)
1510 GOTO 580
1520 FOR A=1 TO 14
1530 CALL SOUND(110+A,1000+A,A)
1540 CALL SCREEN(A)
1550 B=A+2
1560 CALL COLOR(12,B,1)
1570 NEXT A
1580 CALL CLEAR
1590 PRINT SEG$(MSG$,12,55)
1600 GOTO 2230
1610 Y=Y+1
1620 FUEL=FUEL-2
1630 IF Y<32 THEN 1710
1640 CALL GCHAR(X,Y,Z)
1650 IF Z<32 THEN 1520
1660 Y=1
1670 X=X+1
1680 CALL GCHAR(X,Y,Z)
1690 IF Z<32 THEN 1520
1700 FUEL=FUEL-1
1710 Y=Y+1
1720 IF Y<32 THEN 1800
1730 CALL GCHAR(X,Y,Z)
1740 IF Z<32 THEN 1520
1750 FUEL=FUEL-1
1760 Y=1
1770 X=X+1
1780 CALL GCHAR(X,Y,Z)
1790 IF Z<32 THEN 1520
1800 RETURN
1810 CALL HCHAR(X,Y,97)
1820 IF Y>25 THEN 1850
1830 SEC=Y+5
1840 GOTO 1860
1850 SEC=31
1860 FOR LA=Y+1 TO SEC
1870 CALL SOUND(110,110,0)
1880 CALL HCHAR(X,LA,45)
1890 CALL GCHAR(X,LA+1,ZA)
1900 IF ZA=64 THEN 2380
1910 NEXT LA
1920 FOR LA=Y+1 TO SEC
1930 CALL HCHAR(X,LA,32)
1940 NEXT LA
1950 CALL HCHAR(X,Y,32)
1960 RETURN
1970 IF FUEL<=7 THEN 2020
1980 FOR F=32 TO FUEL STEP -1
1990 CALL HCHAR(1,F,32)
2000 NEXT F
2010 GOTO 1250
2020 FOR G=1 TO 14
2030 CALL SCREEN(G)
2040 CALL COLOR(12,G,1)

```

```

2050 CALL SOUND(110,110+G,G*2)
2060 NEXT G
2070 PRINT "YOU RAN OUT OF FUEL "
2080 PRINT
2090 GOTO 2230
2100 FOR A=14 TO 1 STEP -1
2110 CALL SOUND(A+10,1000,A)
2120 CALL COLOR(12,A,1)
2130 NEXT A
2140 PRINT "ALIEN DEFENCE DESTROYED YOUR
SHIP"
2150 GOTO 2230
2160 FOR A=1 TO 14
2170 CALL COLOR(12,A,1)
2180 CALL SCREEN(A+2)
2190 CALL SOUND(100+A,3000+A,A)
2200 NEXT A
2210 SCORE=SCORE+BONUS
2220 PRINT "CONGRATULATIONS! YOU DID YOUR TASK WELL"
2230 PRINT "SCORE=";SCORE+FUEL*10
2240 PRINT
2250 IF SCORE+FUEL*10<=HS THEN 2270
2260 HS=SCORE+FUEL*10
2270 PRINT "HI-SCORE=";HS
2280 PRINT
2290 RA=TIME/SCORE-(FUEL/10)
2300 PRINT "TIME/SCORE(-FUEL BONUS)
RATIO=";RA
2310 PRINT
2320 PRINT "DO YOU WANT ANOTHER GOT?"
2330 INPUT Z$
2340 CALL CLEAR
2350 IF Z$="Y" THEN 20
2360 IF Z$="N" THEN 2370 ELSE 2320
2370 END
2380 SCORE=SCORE+50
2390 CALL HCHAR(XS,32,32)
2400 XS=11
2410 CALL SOUND(5,1000,0)
2420 GOTO 1910
2430 HS=0
2440 CALL CLEAR
2450 PRINT "INPUT YOUR NAME"
2460 INPUT N$
2470 CALL CLEAR
2480 PRINT "*****"
2490 PRINT "*****CAVERN ATTACKER *****"
2500 PRINT
2510 PRINT
2520 PRINT "*****"
2530 PRINT
2540 PRINT "HERE IS YOUR MISSION
COMMANDER ";N$
2550 PRINT
2560 PRINT "USING YOUR ASTRO FIGHTER AND YOUR OWN SKILL, YOU MUST
DESTROY YOUR
ENEMY'S NERVE CENTRE"
2570 PRINT
2580 PRINT "BUT DOING THIS IS NOT AS EASY AS IT SOUNDS. THE NERVE CENTRE(A SMA
LL RED SHIP)
2590 PRINT "HAS LOCATED ITSELF DEEP INTO THE HEART OF THE PLANET"
2600 PRINT
2610 PRINT "PRESS SPACE BAR TO CONTINUE"
2620 CALL KEY(0,K,S)
2630 IF K=32 THEN 2650
2640 GOTO 2620
2650 CALL CLEAR
2660 PRINT "YOU DESTROY IT BY BOMBARDING THE CAVERN WITH YOUR DEADLY BOMBS AND EL
ECTRIFYING
LASER"
2670 PRINT
2680 PRINT "ONCE YOU HAVE MADE A CLEAR PATH BETWEEN YOU AND THE
NERVE CENTRE
YOU HAVE TO
2690 PRINT "BOMB IT TO COMPLETE YOUR
MISSION"
2700 PRINT
2710 PRINT "BUT BEWARE. WHILE YOU ARE
BOMBING YOU MAY ENCOUNTER
OR STALAGMITE
A STALAGMITE
2720 PRINT "OR A DEFENDING ALIEN WITH
A LONG RANGE LASER. THIS
IT SIMPLE AND MAY HELP YOU"
ALIEN IS A B
2730 PRINT
2740 PRINT "PRESS SPACE BAR TO CONTINUE"
2750 CALL KEY(0,K,S)
2760 IF K=32 THEN 2780
2770 GOTO 2750
2780 CALL CLEAR
2790 PRINT "*****HERE ARE YOUR KEYS*****"
2800 PRINT
2810 PRINT "Q-UP"
2820 PRINT "Z-DOWN"
2830 PRINT "SPACE BAR-BOMB"
2840 PRINT "L-LASER"
2850 PRINT "P-HOVER"
2860 PRINT "A-ACCELERATE"
2870 PRINT
2880 PRINT "PRESS SPACE BAR TO CONTINUE"
2890 CALL KEY(0,K,S)
2900 IF K=32 THEN 2920
2910 GOTO 2890
2920 CALL CLEAR
2930 RETURN

```


BUYERS' GUIDE

Languages: Sinclair BASIC. This version is an extended form of the type used in the ZX81. The extensions do not cover structured programming or aid functions, and only deal with the graphics and file handling. This BASIC is very non-standard. **Processor:** Z80A. **Operating System:** Sinclair. **I/O:** none included but the ZX printer can be added without an interface. **Number sold in UK:** 400,000. **Speed:** medium.

Add-Ons: ZX Printer (see ZX81 for description). Microdrives, fast continuous loop tape recorders costing £49.95, need the ZX Interface to be attached. This costs £29.95 when purchased with a Microdrive or £49.95 when bought separately. The interface also provides a printer interface and a networking system. ZX Interface gives two Atari-type joystick ports and a slot for ROM games cartridges. This costs £19.95 with 10 cartridges at present available at £14.95 each. Many other add-ons are available from independent suppliers.

Software: a lot of software is available for almost all applications, particularly games. Despite the business software available, the Spectrum cannot realistically be considered as a business machine.

Opinion: reckoned to be Britain's best-selling computer, when the Spectrum first appeared it was quite an innovation as the lowest priced colour micro. But newer designs have arrived recently with better technical specifications and more useable memory at similar prices. The colour display is not good and a colour display cannot be produced on a few colour TVs. For editing and the method of keyword entry see ZX81. With the Spectrum there are more keywords to be fitted on the same size of keyboard. Sound is through a beeper only. The range of software available is enormous. See also Spectrum 48K (£129.95).

Sinclair Research, Stanhope Road, Camberley, Surrey GU15 3PS
Camberley (0276) 685311



TI-99/4A

£99.95

16K RAM (14.5K useable) 26K ROM

Keyboard: typewriter style, 48 keys, space bar. Two functions per key.

Screen: 32 columns by 24 rows text, 128 characters, all user-definable, 16 colours each for foreground and background. The way colour is handled is rather odd — a colour is assigned to a particular group of characters. There are 16 groups of eight characters, so a particular colour can be assigned, for example, to the letter A but will also affect B to H. High-resolution graphics only available with TI plug-in cartridges.

Sound: three channels of five octaves with noise channel.

Languages: TI BASIC is included, but cannot use machine code, so arcade quality programs cannot be written or bought on cassette for the standard model. To write or use machine code programs, TI's Mini Memory or Editor Assembler cartridges are needed. Machine code can also be used with Extended BASIC, provided 32K is added. LOGO, the teaching language, is also available. **Processor:** TMS 9900. **Operating system:** custom. **I/O:** joystick and cartridge ports. **Number sold in UK:** TI would not comment, our estimate: 50,000. **Speed:** slow.

Add-ons: Most come from Texas Instruments and, apart from joysticks and speech synthesiser, all TI's products need a peripheral expansion box. Available: disc drives, 32K RAM card, RS-232 serial port for printers. TI's own cassette recorder includes a cable; your own will need a special cable. Among the very few independent add-ons is a 32K RAMpack from Arcade, of Manchester, which does not require the expansion module.

Software: More software was becoming available — including US imports — but you may be forced to buy through mail order. TI sells a range of sophisticated cartridge software which is more expensive.

Opinion: IMPORTANT NOTE — Texas Instruments stopped making the TI-99/4A last month after losing money heavily. Now it will manufacture only business micros. TI has promised, however, that it will continue to support the 99/4A. UK companies Arcade, Stainless Software, of Stockport, and Christine Computing, of Watford, have also promised continuing software/add-ons, including imports. The computer is now to be sold at cut prices by dealers to clear remaining stocks. My comment still applies: an interesting computer which is quite good value, even at the pre-announcement price of £99.95. And it looks like a real computer with its good keyboard. Its drawback is that machine code cannot be used with the standard model so there is not much arcade-type cassette software, although this is slowly increasing. However, if all the games that interest you are from TI and you are willing to pay cartridge prices, then this is a good machine. For programmers it has a line editor which, although one of the better versions, is awkward to use. The BASIC is slow but easy to use and comprehensive. Its worst features are that there can be

Tandy MC-10

£99.95

4K RAM 8K ROM

Keyboard: 47 hard plastic keys in Spectrum-style layout, includes 16 graphics keys, space bar. Single key keyword entry.

Screen: 32 columns by 16 lines text, 64 by 32 graphics. Eight colours.

Sound: single channel, five octaves controlled from BASIC.

Languages: Microsoft BASIC. **Processor:** 6803.

Operating system: Microsoft. **I/O:** RS-232. **Number sold in UK:** no comment from Tandy, but small because computer launched recently. **Speed:** medium to fast.

Add-ons: 16K RAM pack. Compatible with most Tandy printers.

Software: small range.

Opinion: Tandy is selling this computer through 220 of its stores, rising to around 300, so support is good. It has been on sale in the US since May, so imported software from independents is likely. No UK independent software is known.

Tandy, Tameway Tower, Bridge Street, Walsall, West Midlands WS1 1LA
0922 648181

Don't let that alien land!

How long can you keep off the marauding alien hordes who are trying to take over your laser base? You'll need a joystick to play Stephen Harris's Space Alien game for the unexpanded TI-99/4A

The alien has landed — a scene from Xtro



What's this? Your scanner is telling you there's an alien landing craft approaching your base — and your experience tells you that it isn't planning a social visit.

You'd better move quickly to make sure that craft doesn't get control of the base. Use your joystick to move the base from side to side, and try to shoot the alien down with your missiles. But don't be too trigger happy, or you'll run out of missiles, and the game'll be over.

You get 10 points for each hit — and a new alien will start to make its way towards your base. How long can you fight them off?

Main variables

QW keeps score
A randomises alien position
F recognises if missile button is pressed
UI recognises if joystick is moved

How it works

100-240 print out instructions
250-390 define characters and colour
400-470 plot stars
480-660 move alien
670-730 move and fire
740-800 move left
810-870 move right
880-920 print laser
930-1060 out of missiles
1070-1160 print out score
1170-1290 end of game

Hints on conversion

CALL CLEAR clears the screen
CALL HCHAR (A,P,G) A = row, P = column, G = positions ASCII character
CALL SCREEN (X) sets screen colour
CALL SOUND (N,V,G) N = length of note, V = frequency, G = volume
CALL KEY (N,V,G) N = console keyboard eg. call key (0,V,G). V = recognises if key is pressed. G = recognises if same key is pressed

```
100 REM SPACE ALIEN
110 REM BY STEPHEN HARRIS
120 REM 1983
130 CALL CLEAR
140 CALL SCREEN(2)
150 FOR ZX=2 TO 16
160 CALL COLOR(ZX,16,1)
170 REM INSTRUCTIONS
180 NEXT ZX
190 PRINT "SPACE ALIEN": "YOU CO
NTROL THE BASE": "AT THE BOTTOM
OF THE SCREEN": "BY USING THE JO
YST STICK. ":
200 PRINT "SHOOT DOWN THE ALIEN
AND": "PREVENT IT FROM LANDING B
Y": "PRESSING THE FIRE BUTTON":
210 PRINT "BUT BEWARE... YOU ONL
Y HAVE": "A LIMITED AMOUNT OF MI
SSILES":
220 PRINT "SCORING=10 POINTS PER
HIT": "NOW PRESS ANY KEY "
230 CALL KEY(0,TYU,UYT)
240 IF (TYU=0)+(TYU=-1) THEN 230
250 CALL CLEAR
260 REM DEFINE CHARACTERS AND
SET UP SCREEN
270 CALL COLOR(14,12,1)
280 CALL CHAR(159,"1818181818181
818")
290 CALL COLOR(16,16,1)
300 D=16
310 Z=2
320 CALL CHAR(139,"1CEE7DFD5FDFF
7D")
330 CALL HCHAR(24,1,139,32)
```


TI-99/4A PROGRAM

```

340 CALL CHAR(42,"1818183C7EFFFF
5A")
350 CALL CHAR(127,"99C3A5FFE7423
C42")
360 CALL COLOR(2,7,1)
370 CALL COLOR(12,8,1)
380 CALL HCHAR(23,D,42)
390 CALL CHAR(149,"0000001000000
000")
400 REM PLOT STARS
410 FOR TU=1 TO 30
420 CALL COLOR(16,16,1)
430 KJ=INT(RND*32)+1
440 PI=INT(RND*24)+1
450 IF (PI=24)+(PI=23) THEN 440
460 CALL HCHAR(PI,KJ,149)
470 NEXT TU
480 REM MOVE ALIEN
490 A=INT(RND*30)
500 IF A>10 THEN 530 ELSE 510
510 A=2
520 GOTO 570
530 IF A>20 THEN 560 ELSE 540
540 A=16
550 GOTO 570
560 A=30
570 CALL HCHAR(Z,A,127)
580 CALL HCHAR(Z,A,32)
590 CALL SOUND(30,-1,0,110,0)
600 Z=Z+1
610 IF Z=23 THEN 620 ELSE 680
620 CALL HCHAR(23,D,127)
630 CALL SOUND(1000,-5,0,990,0)
640 PRINT "THE ALIEN HAS LANDED
ON YOUR BASE"
650 GOTO 1171
660 GOTO 680
670 REM MOVE AND FIRE
680 CALL KEY(1,F,6)
690 CALL COLOR(15,INT(RND*13)+3,
1)
700 IF (F=0)+(F=-1) THEN 710 ELSE
890
710 CALL JOYST(1,UI,AS)
720 IF UI=-4.0 THEN 750 ELSE 730
730 IF UI=4.0 THEN 820 ELSE 490
740 REM LEFT
750 CALL HCHAR(23,D,32)
760 D=D-14
770 IF D<2 THEN 780 ELSE 790
780 D=2
790 CALL HCHAR(23,D,42)
800 GOTO 490
810 REM RIGHT
820 CALL HCHAR(23,D,32)
830 D=D+14
840 IF D>30 THEN 850 ELSE 860

```

```

850 D=30
860 CALL HCHAR(23,D,42)
870 GOTO 490
880 REM PRINT LAZER

890 CALL VCHAR(2,D,159,21)
900 CALL SOUND(30,1000,0)
910 ER=ER+1
920 IF ER>4 THEN 940 ELSE 1030
930 REM OUT OF MISSILES
940 PRINT "YOU RAN OUT OF MISSIL
ES"
950 FOR A=-8 TO -5
960 B=B+110
970 C=10
980 CALL SOUND(500,A,0,B,C)
990 C=C-2
1000 NEXT A
1010 GOTO 1020
1020 GOTO 1171
1030 CALL VCHAR(2,D,32,21)
1040 IF D=A THEN 1080 ELSE 1050
1050 CALL SOUND(30,-7,0)
1060 GOTO 490
1070 REM PRINT OUT SCORE
1080 QW=QW+10
1090 CALL SOUND(100,-7,0)
1100 ER=0
1110 Q$="SCORE="&STR$(QW)
1120 FOR T=1 TO LEN(Q$)
1130 CALL HCHAR(1,T+1,ASC(SEG$(Q
$,T,1)))
1140 NEXT T
1150 Z=2
1151 CALL COLOR(12,INT(RND*14)+3
,1)
1160 GOTO 490
1170 REM END OF GAME
1171 IF QW>AW THEN 1172 ELSE 118
0
1172 AW=QW
1180 PRINT ":: YOUR SCORE IS ";Q
W::
1181 PRINT "HIGH SCORE=";AW
1190 PRINT "PRESS REDD OR BACK"
1200 CALL KEY(0,DI,ED)
1210 QW=0
1220 Z=2
1230 ER=0
1231 B=0
1240 IF (DI=15)+(DI=57) THEN 100
1250 IF (DI=6)+(DI=56) THEN 1270
1260 GOTO 1200
1270 CALL CLEAR
1280 PRINT "* *PRESS ANY KEY TO
BEGIN* *"
1290 GOTO 230

```


LETTERS

The Joy of Joysticks

I have been a TI owner since the end of April this year, and am very grateful to the few magazines, such as yours, which take a real interest in the Texas micro.

However, very few of the programs in magazines such as yours seem to use the Texas joysticks. Though these joysticks are not the ultimate in good design, they can add another dimension or variation to games that would otherwise become boring.

Now before anyone tells me to get off my peripherals, plug in my modules and write a program myself, I'd like to say that I am doing just that — it will be a maze game, using joysticks, and when it has been fully debugged you might just get to see it.

Meanwhile, here is a joystick routine for the Apple Scrumpt game printed in HCW 37. Replace lines 395 to 440 with the following:

```
395 CALL JOYST(1,A,B)
400 CALL HCHAR(Y,X,
32)
405 IF (A<>-4)+(B<>
0) THEN 415
410 X=X-1
415 IF (A<>4)+(B<>0)
THEN 425
420 X=X+1
425 IF (A<>0)+(B<>4)
THEN 435
430 Y=Y-1
435 IF (A<>0)+(B<>
-4) THEN 445
440 Y=Y+1
```

This routine can, with slight modification, be used on many other games.

By the way, I typed this letter myself, in less than 10 minutes, but prior to obtaining my Texas this could have taken anything from two to three months without the practice on a real keyboard.

I doubt this improvement would have been the same using a Spectrum or Oric! I reserve the right to remain a bad speller, though.

Tony "Tonks" Huby,
Bramley, Leeds

Peace on earth — and elsewhere

I appeal to everyone who will be buying, giving or receiving computer games this Christmas to stop and think before making their choices.

I put it to you that encouraging young people to play games the purpose of which is to kill, bomb, shoot, conquer, destroy and show as much aggression as possible is as sick as forcing them to watch "video nasties".

In fact it is sicker, because

Send your letter to Letters, Home Computing Weekly, 145 Charing Cross Road, London WC2H 0EE. Don't forget to name your computer — the best letter could win £5-worth of software. Got a problem with your micro? We'll soon be starting a queries page, so send your technical questions to Queries at the above address



with computer games they must actively participate.

It is a tragedy that while British home micro software leads the world in technical quality, the vast majority of the themes of "games" appear to have been thought up by the minds of mass-murderers to give pleasure to admirers of Adolf Hitler, Jack the Ripper, Attila the Hun and Denis Nilsen.

Look through the adverts in this magazine and feel shame. We have allowed this to happen, and we are all responsible, programmers, advertisers, magazines and shops.

We are using the greatest innovation of our age to teach kids that killing is OK. Pick your presents with care. Happy Christmas. Peace and goodwill to all Mankind, and Aliens.

Mel Croucher, Automata,
Portsmouth, Hants

Computer talk from Norfolk

I seem to have achieved fame — or is it notoriety? — since having a letter published in HCW 31. Not that I'm complaining, as several people took the trouble to get in touch with me — in this neck of the woods, computerising females are a bit thin on the ground, and we need to stick together.

Is there a postal club for adventure game players? If not, would anyone be interested in starting one? I work full time, and don't honestly have the time to run a club, but I would be willing to help.

On the subject of adventure games, has anyone found all seven pieces of parchment in *Invincible Island* (Spectrum 48K)? I've found six, and can't find number seven. Help!!!!

If your child is anything like my daughter, she'll spend hours and hours drawing and colouring patterns. The following two lines help keep them quiet over Christmas!

10 PRINT (Your choice of

graphics, UDGs, spaces);

20 GOTO 10

This gives a screenful of pattern, and can easily be altered by adding or removing shapes or spaces. Enterprising parents can, by adding a few more lines, allow the child to choose paper and ink colours.

On to the complaints section. I've had dealings by post with four companies. Three out of the four have been excellent — Hewson, Gilsoft and Virgin Games. Who gets the booby prize? Quicksilver, that's who. I sent off a cheque to them for their Game Lords Club. It was cashed promptly, but it was over a month before I heard anything from them. Pity. Especially as some of their games are so good.

A. Bates (HCW 38) (No relation as far as I know) is lucky to have got her husband interested in computing. So far, apart from playing the odd game, the most interest mine has shown has been when I put in a simulation of a two-stroke engine.

He liked that, and thought that it might be of some use! Oh well, I shall just have to write a program of motorbike racing!

Best wishes, Happy Christmas, and keep up the good work.

Patricia Bates, North Walsham, Norfolk

Mark Eyles of Quicksilver said: "The Game Lords Club is handled by a professional mail order company, and they're normally very quick — perhaps Mrs Bates's order got held up by the Christmas post. If she has any further problems, she should contact us at Quicksilver directly, and we'll do our best to help. By now, she should have received the first issue of the Games Lord magazine, anyway, which should be some consolation — we think it's really good!"

Cheat the Jetman

Amaze your friends with your high score on Jet Pac, thanks to a slight scoring fault in the program.

On the fourth sheet of Jet Pac there are plane-shaped objects which, when they explode, give you points. If the jetman is placed on the top right hand cloud he cannot be killed.

So if the television is switched off and the computer left on, the program will continue to accumulate a steady score, and if left long enough, any score can be reached without laying a finger on the keyboard!

M. Brunning, Bispham, Blackpool

TI-99/4A CHRISTMAS PROGRAM

This distinctly festive game puts you in a difficult position.

You have been invited to a Christmas party at your boss's home. The mince pies have been made by his wife, and although they're the worst mince pies you've ever come across, your chances of promotion depend on eating as many as you can.

Unfortunately, the pies are so stodgy that you have to drink a glass of sherry before you tackle another one. The sherry has a predictable effect, and your hand becomes increasingly difficult to control.

You can regain control (usually — but not quite always) by drinking a cup of coffee. But asking for coffee is an admission that you've drunk too much — and that costs you an increasing number of points.

If you've eaten at least half the pies on the table, you can pull the cracker. This creates so much excitement that you earn a bonus (which depends on the screen you're on) and the hostess provides you with a new table of more pies which are each worth more than the last lot.

At the same time your "cheer-

fulness" increases, and it may not be so easy to guide your hand the way you want it to go.

There's another problem too.

Greedy grabbing all over the table is definitely not allowed, so your hand can't go back over its past tracks. If it tries to, or if it bumps into the border, you lose points.

It's possible to get trapped in your own tracks. If this happens, or if you cut yourself off from the cracker, you have three chances of asking your hostess to set a new table. The number of opportunities left to do this is shown at the bottom left of the screen, headed "Extras". When you press "H" the hostess pulls the cracker for you and gives you a new table of more challenging pies.

Trying to drink two sherries or to eat two pies in succession is possible but not very good for you. In either case you lose a life, and when you lose five lives the game is over. If this happens, you lose 500 points for not knowing when you've had enough.

Sooner or later your coffees will run out.

Then it's only a matter of time before you become so sodden

The mince pies are so horrible you need a drink to wash them down. And that's only one of the problems on the night your future is on the line. Type Tony Garner's game into your unexpanded TI-99/4A and see if your stomach can stand it

You so p the Chr p

How it works

1 array DIMensioned to hold values of objects plotted on screen
2-3 call initialisation and start
4-15 normal moves
16-31 abnormal moves
32-36 PRINT score or text on top line
37-38 detect key-press and beep
39-40 SAVE present positions of hand
41-53 find desired direction of travel
54-57 provide new screen if one is available and requested
58 call "I've had too much to drink" routine
59-73 provide coffee, empty cup, reduce effects of sherry and deduct penalty points
74-86 adjust player's movements according to influence of sherry
87-96 if player bumps into border or own track, beep and reduce score
97-100 player can't grab cracker until he's eaten at least half the pies
101-106 move hand into a clear space and plot trail
107-125 add 1 to items consumed, remember previous and present item, move hand, decide what kind of trail to leave on next move. Increase sherry-count or score. PRINT new score
126 give obstacle-value to present position
127 call life-lost routine if item consumed is same as last item
129-136 life-lost routine: noise, flashing, removal of token
137-150 no-lives-left routine:

noises, penalty, severe message
151-165 initialise variables
166-171 call initialisation sub-routines: point of entry depends on whether this is the first game or a new game
172-175 print cracker and give it a value
176 prevent pies or glasses landing on player's starting square
177-193 plot pies and glasses, with values. Print hand and initialise its coordinates
194-197 first screen displayed
198 instruction to generate "real" random numbers
199-210 define graphics characters
211-215 set up colours
216-233 PRINT frame for game screen
234-238 PRINT coffee-cups to show coffees left, and record highest print-position
239-243 PRINT hands to show lives left, and record lowest print position
244-251 give protective values to border
252-255 PRINT number of extra screens available
257-298 pull cracker, award bonus points, clear and renew array, alter relevant variables and set up new screen if hand has pulled cracker
299-308 solemn routine if player admits he's had enough to drink. Bonus points and noises
309-311 wait for key-press to play new game
312-322 set up new game, adjust high-score and make witty computer joke



must be polite at boss's Christmas party



Variables

MEM(R,C) the array
F graphics character (hand)
C column coordinate
R row coordinate
PL initial print position (Text and score)
MS score or text for printing
K ASCII code for key pressed
S status of key press (Same as before, different or none)
J general purpose variable for loops
P beep-pitch
V store for R
W store for C
FL counter for extra screens requested
EX original total for extra screens
COF total coffees
CMA column to print new coffee-cup
SH total sherries
SC score
BS bonus points
Z total items consumed
TK graphics character to plot trail
N number of items to be plotted
D value of last item consumed
E value of present item consumed
LIVES lives in hand
LMA column to print life-token
S1,S2,S3,S4 steps by which player's resistance is graduated
REP flag to show a replay has been required
X,Y variables to read numeric DATA
NS variable for string DATA
AL increasing effect of sherries
D (at end of game) a delay counter
HSC high score

with sherry that you lose all control of your hand and start losing points all over the place. There is a way out of this problem, so don't despair.

Press P (for "I've had too much to drink") and, after a short pause which allows you to reflect on the error of your ways, you're given 500 bonus points for honesty, and a chance to play a new game.

Control keys:

Cursor keys (E,S,D,X) to move the hand. Left to itself it moves up the screen

C to ask for coffee (which is deliberately awkwardly placed)

H to go to a new screen by escaping from the present one

P to admit you can't go on

A few hints on tactics: Each new game gives you a character with different powers of resisting the sherry. When the sherry begins to take effect the hand will slow down, and then start moving with a will of its own. Its early moves are down and left.

Unless you're losing too many

points try to get to a new screen before drinking coffee, because this will affect your condition on the next screen. Two or more coffees will have a longer lasting effect than one, but the more screens you survive the more valuable the pies are, so don't waste them on the early screens.

Try not to end up with a mass of coffees or "Extras" or lives left over. Sooner or later it's usually sensible to press P and take the credit.

At present my wife's high score is 18,600: mine is rather lower.

In early stages of the game it's worth going for the cracker when about half the screen is cleared, and then use your "Extras" with discretion. You can rest your hand underneath the cracker without penalty at any time before you've cleared the necessary number of pies.

You can start with a pie or a sherry — but you must remember the last thing you had when you go to a new screen, or else you'll lose a life.

Note: Since the generation of random numbers is not a process which produces real random numbers, it will be possible for players to compare performances directly by always BREAKing and RUNning after the first game. This will produce the same character to control, although his field of operation will be different. But this is a slow business.

Graphics: 152 & 153 holly, 144 berries, 136 & 137 pie and pie-crumbs, 128 & 129 full and empty glass, 120 & 121 full and empty cup, 112-115 hand, 105-108 parts of cracker.

Hints on conversion

The array uses a lot of memory, so much less than 14K of user-available memory will not make conversion possible unless the program is rebuilt with multi-statement lines (which the Texas does not support).

CALL CHAR(number, hex-string) defines the ASCII-coded character with the values in the string

CALL HCHAR(number, number, number, number) plots by row and column as ASCII-coded character (a number of times) horizontally

CALL VCHAR is a vertical version of CALL HCHAR

CALL SOUND should be replaced with any appropriate noises

CALL KEY obtains the ASCII value of any key pressed in variable K

CALL CLEAR clears the screen
CALL COLOR sets up colours by character-set, foreground and background

This program sets up a black background, red glasses, yellow pies, a magenta hand, green holly and a cyan cracker.

TI-99/4A CHRISTMAS PROGRAM

```

1 DIM MEM(24,32)
2 GOSUB 151
3 GOTO 37
4 F=115
5 C=C-1
6 RETURN
7 F=114
8 C=C+1
9 RETURN
10 F=113
11 R=R+1
12 RETURN
13 F=112
14 R=R-1
15 RETURN
16 F=113
17 R=R+1
18 C=C-1
19 RETURN
20 F=113
21 R=R+1
22 C=C+1
23 RETURN
24 F=112
25 R=R-1
26 C=C-1
27 RETURN
28 F=112
29 R=R-1
30 C=C+1
31 RETURN
32 CALL HCHAR(1,PL,32,LEN(M$)+2)
33 FOR J=1 TO LEN(M$)
34 CALL HCHAR(1,PL+J,ASC(SEG$(M$,J,1)))
35 NEXT J
36 RETURN
37 CALL KEY(3,K,S)
38 CALL SOUND(-75,P,0)
39 V=R
40 W=C
41 IF S=0 THEN 52
42 IF K<>83 THEN 45
43 K=1
44 GOTO 74
45 IF K<>68 THEN 48
46 K=2
47 GOTO 74
48 IF K<>88 THEN 51
49 K=3
50 GOTO 74
51 IF K<>69 THEN 54
52 K=4
53 GOTO 74
54 IF (K=72)*(FL<3) THEN 55 ELSE
58

```

```

55 FL=FL+1
56 CALL HCHAR(24,12,ASC(STR$(EX-FL)))
57 GOTO 268
58 IF K=80 THEN 299
59 IF (K=67)*(CDF>0) THEN 60 ELSE
52
60 CALL HCHAR(1,CMAX-2,121)
61 CDF=CDF-1
62 FOR J=220 TO 110 STEP -10
63 CALL SOUND(50,J,0)
64 NEXT J
65 CMAX=CMAX-2
66 IF INT(RND*15)+1=3 THEN 68
67 SH=SH-S2
68 SC=SC-B5
69 IF SC>1 THEN 71
70 SC=0
71 M$=STR$(SC)
72 GOSUB 32
73 GOTO 37
74 IF SH>S1 THEN 77
75 ON K GOSUB 4,7,10,13
76 GOTO 87
77 IF SH>S2 THEN 80
78 ON K+INT(RND*1) GOSUB 4,7,10,13,16
79 GOTO 87
80 IF SH>S3 THEN 83
81 ON K+INT(RND*2) GOSUB 4,7,10,13,16,20
82 GOTO 87
83 IF SH>S4 THEN 86
84 ON K+INT(RND*3) GOSUB 4,7,10,13,16,20,24
85 GOTO 87
86 ON K+INT(RND*4) GOSUB 4,7,10,13,16,20,24,28
87 IF MEM(R,C)<>1 THEN 97
88 R=V
89 C=W
90 CALL SOUND(-100,2200,0)
91 SC=SC-50
92 IF SC>1 THEN 94
93 SC=0
94 M$=STR$(SC)
95 GOSUB 32
96 GOTO 37
97 IF (MEM(R,C)=2)*(Z<N) THEN 98 ELSE 101
98 R=V
99 C=W
100 GOTO 37
101 IF MEM(R,C)<>0 THEN 107
102 CALL HCHAR(V,W,TK)
103 CALL HCHAR(R,C,F)

```


TI-99/4A CHRISTMAS PROGRAM

```

104 TK=145
105 MEM(R,C)=1
106 GOTO 37
107 Z=Z+1
108 IF (MEM(R,C)=2)*(Z)=N) THEN 2
57
109 D=E
110 E=MEM(R,C)
111 CALL HCHAR(V,W,TK)
112 CALL SOUND(-250,-(3+MEM(R,C)
),0)
113 CALL HCHAR(R,C,F)
114 IF E<>3 THEN 117
115 TK=137
116 GOTO 122
117 IF E<>4 THEN 121
118 TK=129
119 SH=SH+1
120 GOTO 122
121 TK=145
122 IF MEM(R,C)<>3 THEN 126
123 SC=SC+BS
124 M$=STR$(SC)
125 GOSUB 32
126 MEM(R,C)=1
127 IF E=D THEN 129
128 GOTO 37
129 LIVES=LIVES-1
130 FOR J=1 TO 10
131 CALL SOUND(200,110,0,220,0,3
00,0)
132 CALL SCREEN(12-J)
133 NEXT J
134 CALL HCHAR(1,LMAX,32)
135 LMAX=LMAX+2
136 IF LIVES<>0 THEN 37
137 FOR J=1760 TO 110 STEP -10
138 CALL SOUND(-100,J,0)
139 NEXT J
140 CALL CLEAR
141 CALL SCREEN(7)
142 SC=SC-500
143 PRINT "SO THAT'S IT . . .":
". . . CHOKED ON A MINCE PIE":
144 PRINT "GREED HAS BEATEN YOU"
:"AND YOU'VE LOST 500 POINTS": "F
OR THE DISGRACE":
145 PRINT "YOUR FINAL SCORE IS":
SC:"POINTS": ". . .AND I HOPE YO
U'RE": "ASHAMED OF YOURSELF":
146 PRINT "SOBER UP AND TRY TO D
O": "BETTER NEXT TIME":
147 FOR J=880 TO 110 STEP -10
148 CALL SOUND(75,J,0)
149 NEXT J
150 GOTO 309
151 BS=50
152 CDF=7
153 FL=0
154 EX=3
155 LIVES=5
156 S1=INT(RND*4)+4
157 S2=INT(RND*5)+S1
158 S3=INT(RND*6)+S2
159 S4=INT(RND*7)+S3
160 P=1250
161 PL=17
162 SC=0
163 SH=0
164 TK=145
165 N=10
166 IF REP<>1 THEN 169
167 GOSUB 216
168 GOTO 170
169 GOSUB 194
170 GOSUB 172
171 RETURN
172 FOR J=1 TO 3
173 CALL HCHAR(13,15+J,103+J)
174 MEM(13,15+J)=2
175 NEXT J
176 MEM(21,5)=1
177 FOR J=1 TO N
178 GOSUB 190
179 CALL HCHAR(R,C,136)
180 MEM(R,C)=3
181 GOSUB 190
182 CALL HCHAR(R,C,128)
183 MEM(R,C)=4
184 NEXT J
185 MEM(21,5)=0
186 CALL HCHAR(21,5,112)
187 R=21
188 C=5
189 RETURN
190 R=INT(RND*22)+2
191 C=INT(RND*30)+3
192 IF MEM(R,C)<>0 THEN 190
193 RETURN
194 CALL CLEAR
195 CALL COLOR(2,16,1)
196 PRINT TAB(3); "THE BOSS'S XMA
S PARTY": TAB(3); "*****
*****": TAB(5); "SETTING UP NO
W": TAB(5); "*****
*****":
197 CALL SCREEN(8)
198 RANDOMIZE
199 DATA 152,01030F0F3E3E78F8,15
3,80C0F0F07C7C1F,144,000C0F333CC
CF020,145,0000180018,136,00007EF
FFF7E7E3C
200 DATA 137,000020A4000D200A,12
8,003E3E3E0808083E,129,0022223E0
808083E,120,003C3F3D3F3CBDF
201 DATA 121,002427252724BDFF,11
2,082AAAAAFC3C3838,113,1C1C7C7F5
5555410,114,1C10FEF0FF303C

```


TI-99/4A CHRISTMAS PROGRAM

```

202 DATA 115,38087F0FFFF0C3C,104,
0004030703070304,105,007E7EFFFFF
F7E7E,106,0020C0E0C0E0C02
203 DATA 107,046242E7C3E34666,10
8,00464AC3C3C34272
204 DATA 16,13,15,7,14,12,13,9,1
2,16,11,14,10,8,8,16,7,16,6,16,5
,16,4,16,3,16,1,16
205 RESTORE 199
206 FOR J=1 TO 19
207 READ X,N$
208 CALL CHAR(X,N$)
209 CALL HCHAR(J,1,32,32)
210 NEXT J
211 RESTORE 204
212 FOR J=1 TO 14
213 READ X,Y
214 CALL COLOR(X,Y,1)
215 NEXT J
216 CALL CLEAR
217 CALL SCREEN(2)
218 FOR J=3 TO 30 STEP 3
219 CALL HCHAR(2,J,152)
220 CALL HCHAR(24,J,144)
221 CALL HCHAR(2,J+1,144)
222 CALL HCHAR(24,J+1,153)
223 CALL HCHAR(2,J+2,153)
224 CALL HCHAR(24,J+2,152)
225 NEXT J
226 FOR J=3 TO 21 STEP 3
227 CALL VCHAR(J,3,144)
228 CALL VCHAR(J,32,144)
229 CALL VCHAR(J+1,3,153)
230 CALL VCHAR(J+1,32,152)
231 CALL VCHAR(J+2,3,152)
232 CALL VCHAR(J+2,32,153)
233 NEXT J
234 FOR J=1 TO 2*CDF STEP 2
235 CALL HCHAR(1,3+J,120)
236 CALL HCHAR(1,4+J,32)
237 NEXT J
238 CMAX=3+J
239 FOR J=1 TO 2*LIVES STEP 2
240 CALL HCHAR(1,22+J,112)
241 CALL HCHAR(1,23+J,32)
242 NEXT J
243 LMAX=23
244 FOR J=3 TO 32
245 MEM(2,J)=1
246 MEM(24,J)=1
247 NEXT J
248 FOR J=2 TO 23
249 MEM(J,3)=1
250 MEM(J,32)=1
251 NEXT J

```

```

252 M$="EXTRAS="&STR$(EX-FL)
253 FOR J=1 TO LEN(M$)
254 CALL HCHAR(24,4+J,ASC(SEG$(M
$,J,1)))
255 NEXT J
256 RETURN
257 CALL HCHAR(V,W,TK)
258 CALL HCHAR(R,C,F)
259 SC=SC+(BS*2)
260 M$="BONUS POINTS! "
261 CALL SOUND(-1000,1760,0)
262 GOSUB 32
263 M$=" "
264 GOSUB 32
265 CALL SOUND(-1000,1860,0)
266 M$=" "&STR$(BS*2)
267 GOSUB 32
268 CALL HCHAR(13,17,107)
269 CALL SCREEN(15)
270 CALL SCREEN(11)
271 CALL SOUND(200,-2,0)
272 CALL SOUND(250,-6,0)
273 CALL SOUND(100,-5,0)
274 CALL SCREEN(7)
275 CALL HCHAR(13,17,108)
276 N=N+10
277 M$="MORE PIES SOON!"
278 GOSUB 32
279 CALL SCREEN(2)
280 FOR R=3 TO 23
281 FOR C=4 TO 31
282 MEM(R,C)=0
283 NEXT C
284 NEXT R
285 IF REP<>1 THEN 287
286 RETURN
287 AL=AL+2
288 SH=(7-CDF)+AL
289 Z=0
290 BS=BS+200
291 P=P+100
292 GOSUB 216
293 IF CDF>5 THEN 295
294 PL=15
295 M$=STR$(SC)
296 GOSUB 32
297 GOSUB 172
298 GOTO 37
299 FOR D=1 TO 1000
300 NEXT D
301 CALL CLEAR

```




CHRISTMAS PROGRAM

```

302 CALL SCREEN(7)
303 SC=SC+500
304 PRINT "SO YOU ADMIT IT..."::"
YOU'VE HAD ENOUGH TO DRINK"::"FO
R BEING SO HONEST YOU GET"::"500
EXTRA POINTS"::
305 PRINT "SO YOUR FINAL SCORE I
S"::SC;"POINTS"::"WHY NOT SOBER
UP..."::"...AND TRY AGAIN?"::
306 FOR J=1760 TO 110 STEP -110
307 CALL SOUND(150,J,0)
308 NEXT J
309 PRINT ::"ANY KEY TO PLAY AG
AIN"::
310 CALL KEY(0,K,S)
311 IF S=0 THEN 310
312 CALL SOUND(150,1280,0)
313 CALL CLEAR
314 IF SC<HSC THEN 316
315 HSC=SC
316 PRINT TAB(2);"I'M CLEARING T
HE TABLE"::"::"(TOP SCORE SO F
AR=";HSC;"")::
317 REP=1
318 E=0
319 GOSUB 280
320 GOSUB 151
321 REP=0
322 GOTO 37
    
```

WE HAVE SOUND ... COLOUR ... ACTION WE HAVE R&R SOFTWARE



Whether you have an Oric, Spectrum or ZX81 you can play golf in the comfort of your home.
ZX81 16K (Not illustrated) All the excitement of the professional circuit. For 1 or 2 players. You can select either 9 or 18 hole course. Try and overcome obstacles such as trees, bunkers, water – even the rough can be tricky! Full on screen instructions. ONLY £3.75

Spectrum Golf 16K Similar to ZX81 Golf but utilizing the Spectrum colour graphics and sound. ONLY £3.75

Oric Golf 48K A Further development of Spectrum Golf utilizing 48K of memory and the Oric graphics. For 1 to 6 players. ONLY £5.95

FOR SPECTRUM

Star Trek 48K See if you can rid the galaxy of the evil Klingons, save the Federation from the treacherous Romulans, race to protect your star-bases from attack. Are you star-ship commander material? ONLY £4.95

Spectipede 16K See how long you can keep the Spectipede at bay as they descend upon you. You'll have the help of poison mushrooms and your gun

which both kill the Spectipedes but look out for the spider – he will try to jump on you. For 1 or 2 players. Operates with most leading Joysticks or the Keyboard. ONLY £5.50

Gnasher 16K The ultimate maze game. Eat the dots but look out for the ghosts they could eat you! If you eat an energy pill you can eat the ghost. Full colour graphics and sound. Operates with most leading Joysticks or the Keyboard. £4.95

NEW FOR THE ORIC

Oric Quizzmaster 48K A menu driven quiz game utilizing sound and colour. Instructions are also given to substitute your own questions and answers. Great fun for all the family and ideal for education i.e. revision.

Quizzmaster 1 – General knowledge, sports and pastimes.

Quizzmaster 2 – General knowledge, rock and pop.

Quizzmaster 3 – Out soon.

ONLY £7.95 each which includes a £2.00 discount voucher for your next Quizzmaster.

Further Quizzmasters to follow.



Dept. HCW 34 Bourton Road,
Tuffley, Gloucester, GL4 0LE
Tel: (0452) 502819

SPECTRUM GOLF
available from
WHSMITH

LETTERS

Apathy that killed the TI

I was greatly dismayed to hear the news of Texas Instruments's withdrawal from the home computer market, and reading Stephen Shaw's letter in HCW 38 did little to light up the gloom.

Unlike many TI owners, I bought my machine fully aware of its limitation, with the belief that Texas would get their marketing act together, thus persuading more prospective micro purchasers to buy a TI-99/4A in order to make the computing press take more notice, software houses to write more software, with the final effect that hardware prices would drop.

Well, soon you'll be able to pick up a TI-99/4A for around £50; unfortunately I don't think anyone will benefit in the long run.

So, TI will honour servicing and repair commitments for the next year or so; that's fine, but anyone would expect as much. What worries me is that I'll never be able to get together the nice little home system that I once hoped to, because soon, the expansion box and card peripherals, never produced in large quantities, will all be snapped up by the more affluent Texas users, before people like me can save enough money.

Stephen Shaw tells us to take out a loan and buy while we can. Unfortunately, those of us who are unemployed (I include myself) are not regarded as good credit risks, for some reason which escapes me...

Seriously, the whole sorry affair leaves me angry. Angry at Texas' terrible marketing, angry at the apathy of TI owners, but I think I reserve my main ire for the computing press (exclude HCW).

I honestly believe that their cold-shouldering of the TI-99/4A is the overriding reason for its downfall; apathy breeds apathy.

I hope you print this, I'd be very interested to hear more views on the business from the many TI users who read your magazine.

Seg Gorham, Wakefield, West Yorks

The invisible guy

In HCW 35 you printed a program about Guy Fawkes, but lines 510 and 520 had a blank space in the quotes. This should have been a graphic "A" so that it could be redefined as a user defined graphic.

I expect this happened because a proper printer was

Send your letter to Letters, Home Computing Weekly, No.1 Golden Square, London W1R 3AB. Don't forget to name your computer — the best letter could win £5-worth of software. Got a problem with your micro? We'll soon be starting a queries page, so send your technical questions to Queries at the above address



used instead of the Sinclair silver paper one.

HCW is the best mag about, I always get it.
A. Whalley, Potters Bar

More praise of Things

I was made redundant last June, and the best buy I ever made was my Spectrum, which has had hours of constant use.

Just for the record, my payoff barely made the hundreds — a week's wages in lieu. I therefore have to count the pennies, and computing mags come well below essentials.

I picked a copy of Home Computing Weekly purely on the strength of a reasonable Spectrum game listing for 38p. It turned out to be nearly a best buy, except for one thing — the reviewer who wrote about Halls of the Things (HCW 39) can't possibly be talking about the game we bought.

Out of the dozen or so games we have begged, borrowed or bought, it is the only one to stay on the computer for more than half an hour. It is the most addictive, highly-competitive and difficult all-action game I have ever had the pleasure of playing.

Up to now, I have played for eight or more hours a day (in between the job-hunting), and my personal best score stands at 630 points. This includes four hard-won rings and 59 monsters spread through three of the eight levels.

I don't think I am alone in liking Hall of the Things. It highlights all the good points of the Spectrum's graphics display, and believe me, at times the little man doesn't move half as fast as I'd like.

One of my friends has bought a Spectrum mainly to play Halls, and another is considering making a pact with a credit shark because his scores are so bad (he doesn't yet own a computer).

My sister-in-law recently made the mistake of buying a 16K Spectrum. Guess what she wants to upgrade it for?

The only things missing from Halls are the screams of dying monsters, or the hiss of flying fireballs as they crisp a "thing". The basic Spectrum is not capable of producing those sounds, so better not try. Leave the silly tunes to the squashed frogs, and play Halls of the Things in silence.

Bill Fox, Matlock, Derbyshire

This is not the first letter we've published in defence of Halls of the Things, and we can only reiterate what our reviewer said last time — our reviews aren't gospel, they're just a personal opinion, though usually a pretty well-informed one. If you disagree with something a reviewer has said, we're always pleased to hear your comments. Sometimes we'll even print them, too!

K-tel: two is better value

K-tel's two-on-one tape was reviewed as two separate products — It's Only Rock 'n' Roll and Tomb of Dracula — in last week's issue.

This was because the review copy arrived without a cassette insert, similar to review versions often sent to magazines, and without paperwork explaining that it was a dual-game tape.

Our reviewer, P.C., says that his opinions of the two games remain the same but points out that, of course, a single tape with two games for £6.95 is much better value for money.

May we repeat our plea to software houses: Send us only tapes in the form in which your customers will buy them.

Our reviewers can only comment on what they receive.

And one company's products will usually be divided among several reviewers. This is impossible if there are several programs on one tape.

Home Computing Weekly

One for the road

I typed Robert King's 3D Road Game (HCW 34) into my VIC-20 and was immediately impressed by the standard of the graphics. I played it for hours. Keep up the good work, HCW!

David Warby, Bognor Regis, West Sussex

Race for your life

Get on your bike for a fast-moving two-player game. Mark Playle wrote Laser Light Bike for the unexpanded TI-99/4A

How it works

100-190 set colours for instructions
200-400 instructions
410-550 define characters
560-640 set colours
650-670 set variables
680-900 draw screen
910-1180 scan keys to move blue bike
1190-1290 move blue bike
1300-1570 scan keys to move red bike
1580-1680 move red bike
1690-1840 explosion, increment score
1850-1980 auto move for blue bike
1990-2120 auto move for red bike
2130-2300 display score, end

Hints on conversion

TI BASIC is fairly standard with two main exceptions. Single statement lines are used and sub-routines for graphics and sound are available, prefixed with CALL, as detailed below.

CALL CLEAR clears screen

CALL SCREEN defines screen colour

CALL CHAR defines a character with a 16-digit hex string from an eight by eight grid. Other machines will use a different method

CALL COLOUR (character set, foreground, background) sets colour of character

CALL HCHAR (row, column, character code, number of repeats) places character on screen at co-ordinates specified and optionally repeats character horizontally

CALL VCHAR similar to HCHAR but repeats character vertically

CALL GCHAR (row, column, variable) similar to a screen PEEK on other machines. Gives the ASCII code at the given screen co-ordinates

CALL SOUND (D,F,V) gives sound of duration D, frequency F and volume V. Three sounds can be produced at once

Main variables

RB, CB red bike, blue bike scores
COU counter to nine
X,Y; X2, Y2 blue bike; red bike
X,Y, positions
E, E2, D, D2 pointers for auto move
B1, B2 character codes for blue, red bikes

You are in a fast-moving duel to the death — it's you or him.

As you speed along your bike leaves a trail. Crash into it, or your opponents, and you lose.

Riding off the grid automatically gives the set to your opponent.

Best out of nine sets wins the game.

Hints on playing: only press the keys when you want to change direction. If you hold down the keys it slows the game, although this is hardly noticeable.

```
100 REM START INSTRUCTIONS
110 CALL CLEAR
120 CALL SCREEN(5)
130 FOR COL=1 TO 8
140 CALL COLOR(COL,16,1)
150 B1=130
160 NEXT COL
170 FOR COL2=9 TO 11
180 CALL COLOR(COL2,8,1)
190 NEXT COL2
200 PRINT "LASER BIKE RACE"
210 PRINT
220 PRINT "THE IDEA OF THIS GAME IS TO"
230 PRINT "AVOID CRASHING INTO YOUR"
240 PRINT "OPPONENT'S TRAIL OR EVEN YOUR OWN."
250 PRINT
260 PRINT "TO MOVE THE BLUE BIKE"
270 PRINT "USE KEYS(A,W,S,Z)"
280 PRINT
290 PRINT "TO MOVE THE RED BIKE"
300 PRINT "USE KEYS(L,P,J,K)"
310 PRINT
320 PRINT "BEST OUT OF NINE WINS!"
330 FOR PRI=1 TO 4
340 PRINT
350 NEXT PRI
360 PRINT "Press the space bar to start"
370 CALL KEY(0,K,S)
380 IF S=0 THEN 370
390 IF K=32 THEN 420 ELSE 370
400 REM END INSTRUCTIONS
410 REM DEFINE CHARS
420 CALL CLEAR
430 CALL CHAR(128,"183C3C183C3C3C18")
440 CALL CHAR(136,"183C3C183C3C3C18")
450 CALL CHAR(129,"183C3C3C183C3C18")
460 CALL CHAR(137,"183C3C3C183C3C18")
470 CALL CHAR(130,"000076FFFF76")
480 CALL CHAR(138,"000076FFFF76")
490 CALL CHAR(131,"00006EFFFF6E")
500 CALL CHAR(139,"00006EFFFF6E")
510 CALL CHAR(132,"FFFFFFFFFFFFFFFF")
520 CALL CHAR(140,"FFFFFFFFFFFFFFFF")
530 CALL CHAR(144,"FF6181818181FF")
540 CALL CHAR(159,"FFFFFFFFFFFFFFFF")
550 REM END DEFINE CHARS
560 REM DEFINE COLOURS
570 CALL COLOR(13,5,16)
580 CALL COLOR(14,10,16)
590 CALL COLOR(15,16,1)
600 CALL COLOR(16,7,1)
610 FOR COL=2 TO 8
620 CALL COLOR(COL,16,7)
630 NEXT COL
640 REM END DEFINE COLOURS
641 REM 0 VARIABLES
650 RB=0
660 CB=0
670 COU=0
680 CALL CLEAR
690 REM SET VARIABLES AND
691 REM DRAW SCREEN
700 X=12
710 Y=3
```

```
720 X2=12
730 Y2=31
740 E=-1
750 E2=1
760 D=0
770 D2=0
780 B1=130
790 B2=139
800 CALL SCREEN(2)
810 CALL HCHAR(1,2,159,30)
820 CALL HCHAR(24,2,159,30)
830 FOR GRID=2 TO 23
840 CALL HCHAR(GRID,2,144,30)
850 NEXT GRID
860 CALL VCHAR(1,2,159,24)
870 CALL VCHAR(1,32,159,24)
880 CALL HCHAR(12,3,130)
890 CALL HCHAR(12,31,139)
900 REM END DRAW SCREEN
910 REM SCAN KEYS FOR BLUE
920 CALL KEY(1,K,S)
930 IF S=0 THEN 1860
940 IF K=4 THEN 980
950 IF K=15 THEN 1030
960 IF K=2 THEN 1080
970 IF K=1 THEN 1130
980 X=X-1
990 D=1
1000 B1=128
1010 E=0
1020 GOTO 1200
1030 X=X+1
1040 D=-1
1050 B1=129
1060 E=0
1070 GOTO 1200
1080 Y=Y+1
1090 E=-1
1100 B1=130
1110 D=0
1120 GOTO 1200
1130 Y=Y-1
1140 E=1
1150 B1=131
1160 D=0
1170 GOTO 1200
1180 REM END SCAN KEYS BLUE
1190 REM MOVE BLUE
1200 EX=0
1210 IF (X=1)+(X=24) THEN 1700
1220 IF (Y=2)+(Y=32) THEN 1700
1230 CALL GCHAR(X,Y,HB)
1240 IF HB=132 THEN 1700
1250 IF HB=140 THEN 1700
1260 CALL HCHAR(X,Y,B1)
1270 CALL HCHAR(X+D,Y+E,132)
1280 GOTO 1310
1290 REM END MOVE BLUE
1300 REM SCAN KEYS FOR RED
1310 CALL KEY(2,K,S)
1320 IF S=0 THEN 2000
1330 IF K=11 THEN 1370
1340 IF K=13 THEN 1420
1350 IF K=17 THEN 1470
1360 IF K=12 THEN 1520
1370 X2=X2-1
1380 B2=136
1390 D2=1
1400 E2=0
1410 GOTO 1590
1420 X2=X2+1
1430 B2=137
1440 D2=-1
1450 E2=0
1460 GOTO 1590
1470 Y2=Y2+1
1480 B2=138
1490 E2=-1
1500 D2=0
1510 GOTO 1590
1520 Y2=Y2-1
1530 B2=139
1540 E2=1
1550 D2=0
1560 GOTO 1590
```


TI-99/4A PROGRAM

```

1570 REM END SCAN KEYS RED
1580 REM MOVE RED
1590 EX=1
1600 IF (X2=1)+(X2=24) THEN 1700
1610 IF (Y2=2)+(Y2=32) THEN 1700
1620 CALL GCHAR(X2,Y2,HB)
1630 IF HB=132 THEN 1700
1640 IF HB=140 THEN 1700
1650 CALL HCHAR(X2,Y2,B2)
1660 CALL HCHAR(X2+D2,Y2+E2,140)
1670 GOTO 920
1680 REM END MOVE RED
1690 REM START BANG+SCORE
1700 CALL CLEAR
1710 COU=COU+1
1720 FOR BANG=0 TO 30
1730 CALL SCREEN(7)
1740 CALL SOUND(100,-7,BANG)
1750 CALL SCREEN(2)
1760 NEXT BANG
1770 IF EX=1 THEN 1810 ELSE 1780
1780 CB=CB+1
1790 IF COU>8 THEN 2140
1800 GOTO 690
1810 RB=RB+1
1820 IF COU>8 THEN 2140
1830 GOTO 690
1840 REM END BANG+SCORE
1850 REM START AUTO MOVE
1860 IF D=1 THEN 1900
1870 IF D=-1 THEN 1920
1880 IF E=1 THEN 1940
1890 IF E=-1 THEN 1960
1900 X=X-1
1910 GOTO 1200
1920 X=X+1
1930 GOTO 1200
1940 Y=Y-1

```

GOTO START GAME
BLUE

```

1950 GOTO 1200
1960 Y=Y+1
1970 GOTO 1200
1980 REM END AUTO MOVE
1990 REM START AUTO MOVE
2000 IF D2=1 THEN 2040
2010 IF D2=-1 THEN 2060
2020 IF E2=1 THEN 2080
2030 IF E2=-1 THEN 2100
2040 X2=X2-1
2050 GOTO 1590
2060 X2=X2+1
2070 GOTO 1590
2080 Y2=Y2-1
2090 GOTO 1590
2100 Y2=Y2+1
2110 GOTO 1590
2120 REM END AUTO MOVE
2130 REM START END GAME
2140 CALL SCREEN(7)
2150 PRINT "SCORE"
2160 PRINT
2170 PRINT "BLUE BIKE SCORED " RB
2180 PRINT
2190 PRINT "RED BIKE SCORED " CB
2200 FOR PRI=1 TO 5
2210 PRINT
2220 NEXT PRI
2230 PRINT "DO YOU WANT ANOTHER GO? (Y/N)"
2240 PRINT
2250 PRINT
2260 CALL KEY(2,K,S)
2270 IF S=0 THEN 2260
2280 IF K=15 THEN 2300
2290 IF K=18 THEN 650 ELSE 2260
2295 REM END GAME
2300 END

```

BLUE
RED

RED
+SCORE DISPLAY



**NEW REVISED
PRICES**

TEXAS INSTRUMENTS SOFTWARE

EXTENDED BASIC£39.50

PARSEC£19.50
MUNCHMAN£18.50
ADVENTURE/PIRATE£15.00
TUNNELS OF DOOM£15.00
CHISHOLM TRAIL£10.50
EARLY READING£12.50

SEND SAE FOR FULL PRICE LIST
PLEASE MAKE CHEQUES/POs
PAYABLE TO

PARCO Electrics

4a Dorset Place
New Street
Honiton, Devon
Tel: 44425



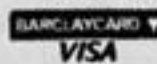
POOLSWINNER

THE ULTIMATE POOLS PREDICTION PROGRAM

- PREDICTS** Not just SCOREDRAWS, but NOSCORES, ALWAYS and HOMES
 - IT WORKS** We **guarantee** the program performs significantly better than chance
 - ADAPTABLE** "Poolswinner" allows the precise prediction formula to be set by the user — you can develop and test your own unique method. Probabilities are given on **every** fixture — choose as many or as few selections as you wish
 - EASY TO USE** Fully menu driven, with detailed instruction booklet
 - DATABASE** The program comes complete with the largest database available — over 20,000 matches. The database automatically updates as results come in
- APPLE, BBC (B), COMMODORE 64, DRAGON, SPECTRUM (48K), ZX81 (16K)
£15 (all inclusive)

We produce databases for those developing their own prediction program. Two years results £7.50. Five years results £12.50

Available from dealers, or direct (return of post) from ...



Selec Software



37 COUNCILLOR LANE, CHEADLE, CHESHIRE 061-428 7425
DEALER ENQUIRIES WELCOME

Put your foot down... but mind those trees

Although he's not old enough for a driving licence, Graham Simpson, aged 12, has produced a challenging driving game for the unexpanded TI-99/4A

You'll need all your driving skill as my game takes you on a forest rally.

Although the program uses just standard TI BASIC, it's a challenge to complete the course.

How many miles will you complete before — crash!?

Although it has the same name, this is not the Forest Rally game in an earlier issue.

How it works

10-380 sets up first screen
390-500 sets up characters and colours
510-560 move car
660-730 reads data for course
740-1920 data for course
1930-1950 starts course again
1960 end routine

Variables

COS strings containing the forest course in the form of P1 to P100 DATA statements
FAR distance in miles car has travelled (score)
C gives position of car in screen column
P provides next front of track to be printed

Hints on conversion

CALL CHAR (ASCII code number, hexadecimal string) defines the pattern to be given to the character with the ASCII code stated

CALL HCHAR (row number, column number, ASCII code, number of repetitions) places character with ASCII stated, at row and column specified, and optionally repeats it horizontally the number of times stated

CALL VCHAR (row number, column number, ASCII code, number of repetitions) works in the same way as CALL HCHAR only repeats vertically

CALL GCHAR (row number, column number, numeric variable) equivalent of PEEK. Assigns ASCII code of character at row and column stated to numeric variable

CALL CLEAR clears the screen

CALL COLOR (character set, foreground colour code, background colour code) specifies foreground and background colours for all characters in set specified

CALL SCREEN (colour code) specifies screen colour using colour code

CALL SOUND (duration, frequency, volume) produces sound. Duration in milliseconds, frequency in Hertz (or noise code), and volume in scale from 0 loudest to 30 quietest

CALL KEY (key unit, k, s) equivalent of INKEY\$, returns code of key pressed in variable k



```
10 DIM COS(120)
20 CALL CHAR(121,"FFFFFFFFFFFFFF")
30 CALL CLEAR
40 CALL SCREEN(5)
50 FOR CL=1 TO 16
60 CALL COLOR(CL,5,1)
70 NEXT CL
80 PRINT "          "
90 PRINT "          "
100 PRINT "          "
110 PRINT "          "
120 PRINT "          "
130 PRINT "          "
140 PRINT "          "
150 PRINT "          "
160 PRINT "          "
170 PRINT "          "
180 PRINT "          "
190 PRINT "          "
200 PRINT "          "
210 P=1
220 RESTORE
230 C=15
240 FAR=0
250 CALL CHAR(120,"304299A1A199423C")
```

```
260 PRINT "      INSTRUCTIONS-PRESS Y"
270 CALL SCREEN(16)
280 CALL KEY(0,K,S)
290 IF S=0 THEN 260
300 CALL SOUND(400,600,0)
310 IF (K<>89)*(K<>121) THEN 340
320 CALL CLEAR
330 GOSUB 2200
340 CALL CLEAR
350 FOR CL=3 TO 8
360 CALL COLOR(CL,14,1)
370 NEXT CL
380 CALL SCREEN(4)
390 CALL CHAR(104,"1010303070701010")
400 CALL CHAR(112,"003A7E5A107E5A00")
410 CALL CHAR(33,"FFFFFFFFFFFFFF")
420 CALL CHAR(40,"0000001010000000")
430 CALL CHAR(41,"0000240000240000")
440 CALL CHAR(42,"005A004242005A00")
450 CALL CHAR(43,"9900001010000099")
460 CALL SCREEN(8)
470 CALL COLOR(10,13,8)
480 CALL COLOR(2,7,1)
490 CALL COLOR(1,6,1)
500 CALL COLOR(11,5,8)
```

```
510 CALL KEY(0,K,S)
520 IF S=0 THEN 640
530 IF K=83 THEN 560
540 IF K=68 THEN 600
550 GOTO 640
560 C=C+1
570 CALL GCHAR(12,C,GET)
580 IF (GET<>33)*(GET<>32) THEN 1960
590 GOTO 640
600 C=C+1
610 CALL GCHAR(12,C,GET)
620 IF (GET<>33)*(GET<>32) THEN 1960
630 GOTO 640
640 CALL HCHAR(12,C,112)
650 CALL GCHAR(13,C,GET)
660 IF P=120 THEN 1930
670 READ COS(P)
680 CALL HCHAR(12,C,33)
690 PRINT COS(P)
700 IF (GET<>33)*(GET<>32) THEN 1960
710 P=P+1
720 FAR=FAR+.01
730 GOTO 510
```


[illegible]

```

1300 RESTORE
1400 PAUSE
1500 GOTO 670
1600 CALL HCHAR(12,C,40)
1700 CALL SOUND(300,-6,4)
1800 CALL HCHAR(12,C,41)
1900 CALL SOUND(400,-6,3)
2000 CALL HCHAR(12,C,42)
2100 CALL SOUND(500,-6,2)
2200 CALL HCHAR(12,C,43)
2300 CALL SOUND(300,-7,0)
2400 FOR DELAY=1 TO 1000
2500 NEXT DELAY
2600 US="GAME OVER"
2700 ROW=12
2800 COLUMN=9
2900 GOSUB 2240
3100 US="YOU TRAVELED "&STR$(FAR)*" MILES"
3110 ROW=15
3120 COLUMN=5
3130 GOSUB 2240
3140 US="PLAY AGAIN-PRESS Y"
3150 ROW=22
3160 COLUMN=9
3170 GOSUB 2240
3180 CALL KEY(0,K,S)
3190 IF S=0 THEN 2180
3200 CALL SOUND(400,600,0)
3210 IF K=89 THEN 10
3220 CALL CLEAR
3230 END
3240 FOR K=1 TO LEN(US)
3250 CALL HCHAR(ROW,COLUMN+K,ASC(SEG$(US,K,1)))
3260 NEXT K
3270 RETURN
3280 FOR CL=1 TO 16
3290 CALL COLOR(CL,5,1)
3300 NEXT CL
3310 CALL SCREEN(5)
3320 PRINT "
3330 PRINT "          FORREST RALLY"
3340 PRINT "          X 1983 GRAHAM SIMPSON"
3350 PRINT "          MOVE YOUR CAR USING KEYS"
3360 PRINT "          PRESS ALPHA-LOCK KEY DOWN"
3370 PRINT "          PRESS ANY KEY TO PLAY"
3380 CALL SCREEN(0)
3390 CALL KEY(0,K,S)
3400 IF S=0 THEN 3380
3410 CALL SOUND(500,600,0)
3420 RETURN

```



**DO
YOU
HAVE
A
DRAWER
FULL OF
UNWANTED
SOFTWARE?**

Then **SWAP** your unwanted Software by using our Computerised Exchange Service.

For a fee of only £1.40 per cassette exchange, send us your name and address plus a list of swappable cassettes (titles you have, titles you require and for which computers), and we'll do the rest!

Write to (no stamp required): TEL FLEET
(02514) 6624



Freepost, Fleet, Aldershot
Hants GU13 9BR.

N.B. All Software must be in original packing and in good condition.

- No Membership required.
- Exchange as many and as often as you like.
- Send no money now!

COMMODORE 64 PROGRAMMING

```

1380 DATA133,1,173,14,220,9,1,141,14,220,96
1390 DATA173,72,3,145,80,160,40,145,80
1400 DATA169,,162,,133,82,133,83
1410 DATA164,82,177,251,164,83,145,253
1420 DATA200,145,253,230,82,230,83
1430 DATA230,83,192,7,208,235,165,253
1440 DATA24,105,64,144,3,24,230,254
1450 DATA230,254,133,253,232,169,,224,2
1460 DATA208,212,24,144,179
2000 DATA32,126,124,226,123,97,255,236
2010 DATA108,127,225,251,98,252,254,160
2020 DATA,,6,,,,,49,234,,,,,
2030 DATA134,30,142,24,150,139,159,126
2040 DATA168,250,179,6,189,172,200,243
2050 DATA212,230,225,143,238,248,253,46
2100 DATA32,119,200,165,251,133,80
2110 DATA32,119,200,165,251,24,170
2120 DATA164,80,76,240,255
2130 DATA32,119,200,165,251,41,7,133,81
2140 DATA32,119,200,165,251,170,165,81
2150 DATA208,18,138,240,2,169,8,133,82
2160 DATA173,22,208,41,247,5,82
2170 DATA141,22,208,96,201,1,208,18
2180 DATA138,240,2,169,8,133,82
2190 DATA173,17,208,41,247,5,82
2200 DATA141,17,208,96,201,2,208,16
2210 DATA138,41,7,133,82,173,22,208
2220 DATA41,248,5,82,141,22,208,96
2230 DATA201,3,208,16,138,41,7,133,82
2240 DATA173,17,208,41,248,5,82
2250 DATA141,17,208,96,201,4,208,18
2260 DATA138,240,2,169,16,133,82
2270 DATA173,17,208,41,239,5,82
2280 DATA141,17,208,96,201,5,208,5
2290 DATA138,141,24,212,96,201,6,208,5
2300 DATA138,141,5,220,96,138,141,138,2
2310 DATA96

```

```

520 SYSGTEXT,X,1,1,2,C
530 NEXT
590 FORN=0TO999:NEXT
600 SYSMODE,6,0,0:PRINT"J"
610 SYSGCOL,5,0:SYSGCOL,6,0
620 SYSGCOL,1,4
630 FORN=4TO315STEP4:SYSTPO,0,N,100:NEXT
640 SYSGCOL,1,8
650 FORN=0TO199STEP4
660 SYSTPO,0,4,N:SYSTPO,0,315,N
670 NEXT
680 SYSGCOL,1,6
690 FORN=8TO311STEP4
700 S=100-90*SIN((N-8)*PI/152):SYSTPO,0,N,S
710 NEXT
720 SYSGCOL,1,2
730 FORN=8TO311STEP4
740 S=100-90*COS((N-8)*PI/152):SYSTPO,0,N,S
750 NEXT
790 FORN=0TO999:NEXT
800 PRINT"J"
810 FORN=0TO39
820 Y%=12-12*SIN(N*PI/20):SYSGCOL,4,N
830 SYSCRSR,N,Y%:PRINT"HELLO"

```

```

840 NEXT
920 M=15
925 X=32
930 FOR0=7TO2STEP-1
940 FORS=23TO0STEP-2:T=5/(S+3)
944 SYSMUSIC,0,X,0,0,0,0
945 L=0-1:SYSMUSIC,1,X,0,0,L,0
947 FORTT=0TO99:NEXT
950 K%=S/2:SYSMUSIC,0,X,15,K%,0,0
951 L=0-1:SYSMUSIC,1,X,15,K%,L,0
970 FORN=0TO15STEPT:V%=15-N:SYSVOL,V%:NEXTN,S,0
980 SYSVOL,0

```

Listing 2 — a demonstration program shows the versatility of the new commands

```

10 MODE=51828:GCOL=51770:PO=51334
20 GMOVE=52039:LINE=51754:TP0=51501
30 CLG=52151:GTEXT=52657:CRSR=53056
40 SPRITE=52164:SETSPRITE=52248
50 MUSIC=52453:VOL=52646:FX=53075
80 SYSGCOL,5,2:SYSGCOL,4,6
98 REM*IF YOU DO NOT HAVE PART ONE*
99 REM*GOTO290*
100 SYSMODE,0,0,1
110 SYSGCOL,1,0
120 FORN=0TO319STEP2
130 Y%=100-80*SIN(N*PI/160):SYSP0,2,N,Y%
140 NEXT
150 SYSGCOL,1,7
160 FORN=0TO199STEP6
170 SYSGMOVE,30,100:SYSLINE,2,300,N
180 NEXT
190 SYSGCOL,0,8
200 FORN=0TO199STEP6
210 SYSGMOVE,310,100:SYSLINE,6,30,N
220 NEXT
290 SYSMODE,0,0,0
300 A(0)=8:A(1)=5:A(2)=12:A(3)=12:A(4)=15
310 FORN=0TO4
320 SYSGTEXT,N,0,235,0,A(N)
330 SYSGTEXT,N,1,5,2,A(N)
340 NEXT
400 FORN=0TO4
410 X=N+5:C=A(N)+256:SYSGTEXT,X,0,114,0,C
420 SYSGTEXT,X,1,39,2,C
430 NEXT
500 FORN=0TO4
510 X=N+10:C=A(N)+128:SYSGTEXT,X,0,16,0,C

```

Micro Tip

TI-99/4A

A question of definition

Home Computing Weekly has printed a number of programs for the VIC-20 in two or even three parts. Typically, the first program defines the graphics and then a second program uses them.

So, what can the TI-99/4A do?

The standard character set is always restored when a program is completed and another loaded, so we can't use those.

In TI BASIC, the cassette loader uses the same area of VDP RAM as user defined graphics. Loading a tape immediately removes your definitions for these characters, so that's out.

Hmm... Extended Basic uses different memory mapping! If we write a small program to define characters 127 to 143, and then load a second program from tape, the redefined characters can be used.

That isn't very many characters, of course, but if memory is really tight, using an initial character defining program will save over 300 bytes of memory at minimum, and possibly as much as 500 bytes. That can make a difference sometimes.

Try it. In Extended BASIC enter and run:

```
10 FOR T=127 TO 143::CALL CHAR(T,"F1010101F1")
::NEXT T
```

Now load a tape program which uses (but does not define) characters 127 to 143. You will see they have retained their definitions from the first program.

If you do not have such a program, try something simple like:

```
10 FOR T=127 TO 143::PRINT T;CHR$(T)::NEXT T
```

Stephen Shaw

Griddlebug/ Bogeyman TI-99/4A £7.95

Softstuff, PO Box 182, Cambridge

I can't help liking a game that rewards a high score with "Oh I say, jolly well done!" It makes a refreshing change from the usual jargon.

Griddlebug is fast moving, calling for quick responses. You have to keep griddlebug alive by guiding him to meals and avoiding the voracious Gridmaster by dodging behind Golden Apples or Power Packs. You must size up the situation quickly — delay is fatal. Success takes you to the next screen for more of the same.

The game is very difficult but persevere and you may discover

Gridmaster's weaknesses, then life gets easier. There are 10 levels — I gave up at three. The set-up routine is tiresome, otherwise a very good game.

I cannot say the same for Bogeyman. Your task is to fight germs causing Clive's cold — yes, that sort of Bogeyman. It is difficult to control with very poor key response. Scoring system is incomprehensible and the game must be RUN for each play. The instructions go on in a humorous way, which is fun but makes things less than clear. There are no on-screen instructions for either.

instructions	30%
playability	70%
graphics	80%
value for money	75%



Countdown VIC20 + 8K/16K £5.50

Paramount, 67 Bishopton Lane, Stockton, Cleveland

This is a goodie. Once you realise your best chance of survival, and high score, is to keep moving, it shows all the ingredients of a really addictive game.

The scene is the countdown to a bomb going off. You must get out of the house (screen one), through the jungle (screen two) and down the sewers (screen three) where you find and defuse the bomb. If you succeed you are straight back to screen one for another go. Each screen has its quota or hazards, firing bolts, dropping coconuts and so on to be avoided. Unusually the first screen is the hardest.

With three screens, the

programmer has gone to town on the graphics and colours. It shows what can be done with an expanded VIC. There are three skill levels but, rather than speeding up the action, it cuts down defusing time.

My main complaint is that it only loaded properly about once in 10 attempts. I know it could just be my copy, but with an increasing degree of copy protection we seem to get increasing unreliability of loading. Sort it out fellows. No one wants even a brilliant game if it's too much of a pain to load.

B.J.

instructions	85%
playability	95%
graphics	92%
value for money	90%



3-D Race TI-99/4A £6

Stainless Software, 10 Alstone Road, Stockport, Cheshire SK4 5AH

You are at the wheel of a high speed racer and you must overtake five other cars plus a ghost car which appears from time to time. After selecting a skill level (one — six), the five cars appear on the track and zoom off. You select first gear and accelerate after them. You can change lanes, accelerate and brake.

Even at level one the game turned out to be pretty difficult. Although instructions accompany the tape I found difficulty at first in understanding what was going on, in particular which car I

was supposed to be driving. However, after several plays I got the idea and started enjoying the game. Speed, time, temperature and fuel are at the bottom of the screen. Graphics, giving a drivers eye view of the track, are excellent.

There are some pretty sophisticated car racing games on the market and, to be fair, 3-D Race does not come up to their standard but then it is written in BASIC (it has to be for the standard TI) and the programmers have used the language extremely well. All in all, good value for money.

D.B.

instructions	80%
playability	80%
graphics	90%
value for money	90%



From cars to cruisers

Newly-launched games for a selection of computers get the once over from our review panel

Picture Puzzles Dragon 32 £5.95

Lyversoft, 66 Lime Street, Liverpool

You move them about using cursor keys spelled "curser" in the instructions! and a rather surprising release from a company which seems to specialise in fast action games. This has no action at all, being a sliding block puzzle program.

You must remember the sort of thing — you have a picture surrounded by a frame and one part is empty so that you can rearrange the picture and then try to reform it again.

This program contains six pictures each in 15 pieces and they are of a reasonably high graphic

press ENTER to make the move. You are shown the initial puzzle and then you observe the scrambling process. You can even create your own pictures if you so desire and have the time to spend.

What I still find hard to accept is that anyone will want to pay this amount of money for something which in its original form cost pennies. If it's your thing then it will probably satisfy you. It certainly does all it claims but it bores me silly.

D.C.

instructions	60%
playability	60%
graphics	80%
value for money	60%



Cruiser/ Character Generator TI-99/4A £5.55

Solway Software, 6 Curzon Street, Maryport, Cumbria CA15 6LL

In Cruiser you go on a long pub crawl round a canal seeking refreshment at each of the lockside pubs. It doesn't matter if you're not a drinker as you may either collect beer or lemonade!

Using the arrow keys, you steer round the waterway starting from a marina at the top. The course becomes gradually more blocked as other boats appear and moor. Crashing into these too often results in a long walk home or a soaking.

You are given a bot load of

provisions, which gradually become depleted. Re-stocking can only be achieved by returning to the starting point.

The lockside pubs close after serving each pint, so during your journey they must be regularly re-opened to continue scoring.

Overall, amusing with plenty of variety, skill levels run from 1 to 30, well crashproofed and even an option for users with black and white TVs.

Side 2 has a useful character generator for defining your own characters and obtaining their hexadecimal codes.

J.W.

instructions	100%
playability	75%
graphics	80%
value for money	100%



Fed up with slow TI games? This one's really fast

How it works

10-90 initialises variables, set up screen, print titles
100-310 main loop to move copter
320-350 check if copter has hit building
360-550 screen set up, define characters, set colours
570-610 print buildings in random positions
640-650 set up heading, GOSUB to print it
690-790 FOR...NEXT loop for bomb etc
810 explosion if copter hits building
820-830 print explosion
880-890 check if bomb has hit a tower
910-920 bomb explodes on tower
1000-1210 copter crashed — game ends
1220-1250 print headings
1300-1500 success — game ends
1350 calculate score dependent on level and bombs left
1420-1440 bonus score
1760-1820 high-score routine

Oh, no! you may say — not another version of Bomber. But this one is different.

I was fed up with slow BASIC games and since I liked the idea of Bomber I decided to write my own version and to make it run much faster than those I had seen.

Clever programming has squeezed more speed out of the standard TI-99/4A. Gareth Thomas explains what he did and how to play the game

```
10 REM  CRAZY COPTER
20 REM  FOR H.C.W
30 REM  BY GARETH THOMAS
40 DEF RD(A)=INT(RND*A)+1
60 BST=0
70 N$="G.T"
80 SC=0
90 GOSUB 1520
100 GOSUB 370
110 GOSUB 140
120 REM  MAIN LOOP
130 REM  MOVE COPTER
140 CALL KEY(0,K,S)
150 IF S=0 THEN 140
151 CALL HCHAR(4,2,32,28)
160 FOR X2=2 TO 22
170 FOR Y2=3 TO 30
180 CALL KEY(0,K,S)
190 IF S=1 THEN 200 ELSE 220
200 IF BMS<1 THEN 220
210 GOSUB 670
220 CALL HCHAR(X2,Y2-1,32)
230 CALL HCHAR(X2,Y2,103)
240 IF X2>12 THEN 330
```

```
250 NEXT Y2
260 CALL HCHAR(X2,Y2-1,32)
290 NEXT X2
300 GOSUB 1260
310 GOTO 80
320 REM  CHECK FOR HIT
330 CALL GCHAR(X2,Y2+1,A)
340 IF A=151 THEN 360
350 GOTO 250
360 GOSUB 810
370 RANDOMIZE
380 REM  SET UP SCREEN
390 CALL CLEAR
400 CALL COLOR(16,1,1)
410 CALL COLOR(14,2,5)
420 CALL SCREEN(6)
430 CALL VCHAR(1,32,155,24)
440 CALL VCHAR(1,1,155,24)
450 CALL VCHAR(1,31,155,24)
460 CALL COLOR(1,5,5)
470 CALL COLOR(15,2,16)
480 CALL COLOR(13,3,5)
490 CALL COLOR(10,14,5)
500 CALL COLOR(9,11,5)
510 CALL CHAR(151,"FF95FFA9FFA5FFA1")
520 CALL CHAR(129,"FFFFFFFF00000000")
530 CALL CHAR(143,"0000000000000010")
540 CALL CHAR(104,"49AA3C5F389310")
550 CALL CHAR(103,"007F089EFD1E")
560 REM  PRINT BUILDINGS
570 CALL HCHAR(23,3,129,27)
580 FOR G=3 TO 29
590 A=RD(8)
600 CALL VCHAR(23-A,G,151,A)
```

```
610 NEXT G
620 X=1
630 Y=3
631 PS=0
640 M$="C R A Z Y C O P T E R "
650 GOSUB 1220
651 PS=PS+3
652 M$="press any key to start "
653 GOSUB 1220
660 RETURN
670 BMS=BMS-1
680 REM  DROP BOMB
690 FOR X3=X2+2 TO RD(10)+11
700 IF X3>12 THEN 880 ELSE 710
710 CALL HCHAR(X3-1,Y2-1,32)
720 CALL HCHAR(X3,Y2-1,143)
730 NEXT X3
740 CALL SOUND(-600,-6,0)
750 CALL HCHAR(X3-1,Y2-1,104)
760 FOR D=1 TO 100
770 NEXT D
780 CALL HCHAR(X3-1,Y2-1,32)
790 CALL VCHAR(X3,Y2-1,32)
800 RETURN
810 CALL SOUND(1500,-7,0)
820 CALL HCHAR(X2,Y2,104)
830 CALL HCHAR(X2,Y2+1,32)
840 FOR D=1 TO 400
850 NEXT D
860 CALL HCHAR(X2,Y2,32)
870 GOSUB 1000
880 CALL GCHAR(X3,Y2-1,B)
890 IF B=151 THEN 910
900 GOTO 710
910 CALL SOUND(600,-6,0)
```

The result even surprised me as it's quite fast.

I have used many techniques to speed up the program, using as few IF...THEN statements as possible and only one random number generator (as a user-defined function).

Nested FOR...NEXT loops move the copter, eliminating the need to test whether it is off the screen. These are also much faster than C=C+1 and IF C=10 THEN and so on.

The program is also structured as far as is possible.

The game itself is slightly different from the familiar version:

You are carrying thermodynamic bombs to an air force base when you start to lose height off the city and have to bomb the buildings to clear a safe landing area.

But the bombs are old and unstable and may explode without hitting anything... which can be quite frustrating.

Sometimes a building will have to be bombed twice to destroy it totally.

Any key may be used to drop the bombs, there is a bonus system if a screen is cleared, a high-score routine and a level selection at the beginning.

Good luck.

TI-99/4A PROGRAM

```

920 CALL HCHAR(X3-1,Y2-1,104)
930 FOR D=1 TO 100
940 NEXT D
950 ER=RD(5)
960 CALL VCHAR(X3,Y2-1,32,ER)
970 SC=SC+ER
980 CALL HCHAR(X3-1,Y2-1,32)
990 RETURN

1000 CALL SOUND(500,150,0)
1010 CALL SOUND(1000,110,0)
1020 GOSUB 1030
1030 CALL CLEAR
1040 CALL SCREEN(2)
1050 FOR COL=1 TO 9
1060 CALL COLOR(COL,16,2)
1070 NEXT COL
1080 SC=(SC+BMS)*L
1090 IF SC>BST THEN 1100 ELSE 1120
1100 BST=SC
1110 GOSUB 1760
1120 PRINT TAB(2);"BAD LUCK YOU CRASHED":
1130 PRINT TAB(2);"YOU HAD";BMS;"BOMBS LEFT":
1140 PRINT TAB(4);"YOU SCORED ";SC
1150 PRINT :
1160 PRINT "BEST SCORE IS";BST;"BY ";N$
1170 PRINT :
1180 PRINT "PRESS ANY KEY FOR NEW GAME":
1190 CALL KEY(0,K,S)
1200 IF S=0 THEN 1190
1210 GOTO 80
1220 FOR T=1 TO LEN(M$)
1230 CALL HCHAR(PS+1,Y+T,ASC(SEG$(M$,T,1)))
1240 NEXT T
1250 RETURN
1260 FOR T=1 TO 10
1270 READ P
1280 CALL SOUND(-200,P,0,P+1,0)
1290 NEXT T
1300 CALL CLEAR
1310 CALL SCREEN(2)
1320 FOR COL=1 TO 9
1330 CALL COLOR(COL,16,2)
1340 NEXT COL
1350 SC=(SC+BMS)*L
1360 IF SC>BST THEN 1370 ELSE 1380
1370 GOSUB 1760
1380 PRINT TAB(2);"WELL DONE YOU'VE":
1390 PRINT TAB(2);"MADE IT ":
1400 PRINT TAB(2);"YOU SCORED";SC
1410 PRINT :
1420 BON=RD(500)+100
1430 PRINT TAB(2);"YOU RECEIVE A BONUS OF";BON
1440 SC=SC+BON
1450 BST=SC
1460 PRINT :
1470 PRINT TAB(2);"PRESS ANY KEY TO PLAY AGAIN":
1480 CALL KEY(0,K,S)
1490 IF S=0 THEN 1480
1500 RETURN
1510 DATA 300,400,500,600,500,400,300,200,600,200
1520 CALL CLEAR
1530 CALL SCREEN(2)
1540 FOR COL=1 TO 9
1550 CALL COLOR(COL,8,2)
1560 NEXT COL
1570 PRINT TAB(4);"C R A Z Y   C O P T E R   ":
1580 PRINT TAB(3);"PRESS:-":
1590 PRINT TAB(4);"1)FOR 200 BOMBS":
1600 PRINT TAB(4);"2)FOR 150 BOMBS":
1610 PRINT TAB(4);"3)FOR 100 BOMBS":
1620 PRINT TAB(4);"4)FOR 75 BOMBS":
1630 CALL KEY(0,K,S)
1640 IF S=0 THEN 1630
1650 L=K-48
1660 IF (L<1)+(L>4)THEN 1630
1670 ON L GOTO 1680,1700,1720,1740
1680 BMS=200
1690 RETURN
1700 BMS=150
1710 RETURN
1720 BMS=100
1730 RETURN

```



```

1740 REM HIGH SCORE
1750 RETURN
1760 CALL SOUND(1000,300,0,400,0,500,0)
1770 PRINT TAB(2);"WELL DONE!!":
1780 PRINT TAB(2);"YOU HAVE TODAY'S BEST SCORE":
1790 PRINT TAB(2);"PLEASE ENTER YOUR NAME":
1800 INPUT N$
1810 PRINT :
1820 RETURN

```

**CHRISTINE
COMPUTING Ltd**
6 Florence Close, Watford, Herts.
(0923) 672941

TI-99/4A HARDWARE

32K Stand alone ram	£125
Arcade quality joystick	£22
Cassette interface leads - single	£5.40
- dual	£6.90
Compatible cassette recorder	£25.95
Rec + single lead	£30.95
Rec + dual lead	£31.95
Atari/Commodore to TI joystick interfaces	
single	£6
dual	£8.50

SOFTWARE

We are able to offer our own range as well as selected items from Stainless Software, Tomputer, Fantasia 99 and many more.

For full list send 50p (refund 1st order) or cheque/PO with order to above address. Please add 50p p&p to any order under £10.

THANK YOU.

falling fruit! Watch out for your glassware

There's never been a game with hazards like this: cherries, ice cubes and olives. G. W. Hartley wrote it for the standard TI-99/4A

Save the glasses from the fruits that are falling or some of your best crystal will be smashed.

This program runs in TI BASIC with no expansion needed. The object is to catch all the fruit that is descending onto your glasses.

Variables

HS holds the name of the player with high score
SC score
CV bats left
X horizontal position of bat
A horizontal position of fruit
MS holds the message for PRINT AT routine

How it works

100-150 REMs
160-230 clear screen, set colours
240-310 define characters
320-400 sets variables
420-550 set up red bands on screen
610-680 set variable MS to the screen messages
690-830 movement of fruit and bat
840-890 lose a life routine
900-1060 win points and change fruits
1070-1100 set MS to "Game over"
1110-1120 time delay
1130 clear screen
1140 check to see if hi-score is beaten
1180-1220 print congratulations message, asks for player's name
1290-1320 PRINT AT routine
1330-1520 display instructions
1530-1540 wait until a key is pressed
1570-1680 ask for level of play

The game starts by asking you for the level (1-3), then the screen is set up with seven glasses on the bottom band.

You must start to catch the fruits. These include cherries, ice cubes and olives.

The fruit that will drop first will be the cherry, then at 140 points the ice cubes will fall and at 900 points the olives.

You have four lives to waste at the start of the game. So good luck...

Hints on conversion

TI BASIC is fairly standard with two main exceptions. Single statement lines are used and subroutines for graphics and sound are available, prefixed with CALL, as detailed below.

CALL CLEAR clears screen

CALL SCREEN defines screen colour

CALL CHAR defines a character with a 16-digit hex string from an eight by eight grid. Other machines will use a different method

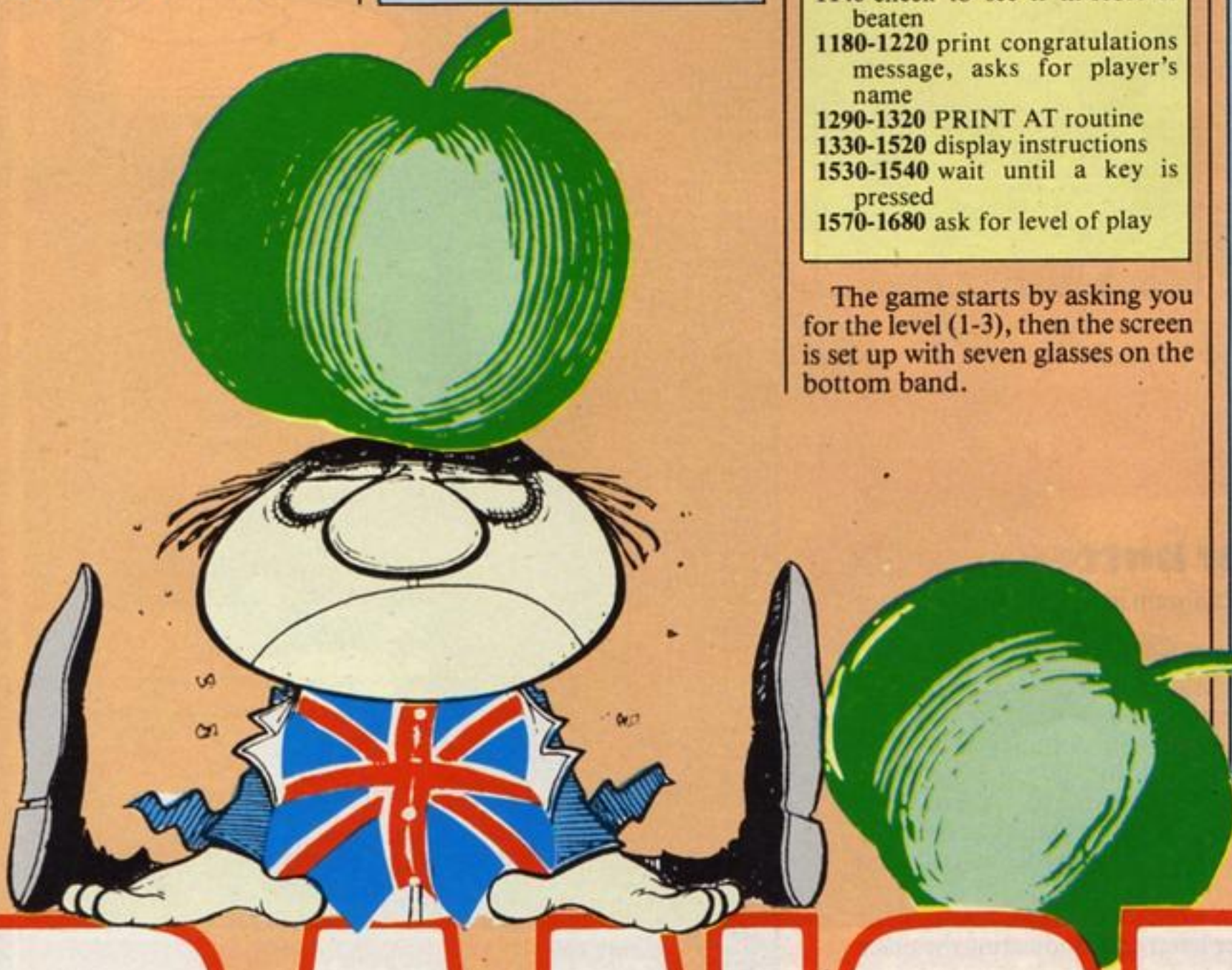
CALL COLOUR (character set, foreground, background) sets colour of character

CALL HCHAR (row, column, character code, number of repeats) places character on screen at co-ordinates specified and optionally repeats character horizontally

CALL VCHAR similar to HCHAR but repeats character vertically

CALL GCHAR (row, column, variable) similar to a screen PEEK on other machines. Gives the ASCII code at the given screen co-ordinates

CALL SOUND (D,F,V) gives sound of duration D, frequency F and volume V. Three sounds can be produced at



TI-99/4A PROGRAM

```

100 REM *****
110 REM * FRUIT COCKTAIL *
120 REM *****
130 REM
140 REM BY G.W.HARTLEY
150 REM
160 CALL CLEAR
170 CALL SCREEN(2)
180 FOR I=1 TO 12
190 CALL COLOR(1,16,2)
200 NEXT I
210 CALL COLOR(13,9,2)
220 CALL COLOR(14,16,2)
230 CALL COLOR(15,9,2)
240 CALL CHAR(129,"FFFFFFFFFFFFFF")
250 CALL CHAR(136,"3F3F1F0F07030101")
260 CALL CHAR(137,"FCFC8F0E0C000000")
270 CALL CHAR(138,"0101010101010101")
280 CALL CHAR(144,"0007040070F0F060")
290 CALL CHAR(139,"00000000000F0FC")
300 CALL CHAR(148,"8567756836370975")
310 CALL CHAR(141,"0101FF")
320 SH="0003040070F0F060"
330 H$="77777777"
340 GOSUB 1330
350 CALL CHAR(144,SH)
360 CALL COLOR(15,9,2)
370 GOSUB 1570
380 SC=0
390 CV=4
400 X=16
410 REM
420 FOR N=2 TO LN
430 CALL HCHAR(N,1,120,32)
440 NEXT N
450 FOR N=18 TO 23
460 CALL HCHAR(N,1,120,32)
470 NEXT N
480 P=9
490 FOR I=1 TO 7
500 CALL HCHAR(16,P,136)
510 CALL HCHAR(17,P,130)
520 CALL HCHAR(16,P+1,137)
530 CALL HCHAR(17,P+1,139)
540 P=P+2
550 NEXT I
560 REM
570 M$="HI-SCORE BY:--"&H$
580 II=24
590 OA=6
600 GOSUB 1290
610 M$="HI-SCORE "&STR$(HS)
620 II=1
630 OA=18
640 GOSUB 1290
650 M$="SCORE"&STR$(SC)
660 II=1
670 OA=2
680 GOSUB 1290
690 RANDOMIZE
700 R=INT(RND*11)+9
710 FOR I=LN+2 TO 14
720 CALL KEY(0,R1,R2)
730 IF R1<>R2 THEN 750
740 X=X-1

```

```

750 IF R1<>R2 THEN 770
760 X=X+1
770 CALL VCHAR(1,A,144)
780 CALL VCHAR(1-1,A,32)
790 CALL HCHAR(15,X-1,32)
800 CALL HCHAR(15,X,141)
810 CALL HCHAR(15,X+1,32)
820 IF (I=14)&(A=X) THEN 900
830 NEXT I
840 CALL SOUND(450,-6,2)
850 CALL HCHAR(16,A,140)
860 CALL HCHAR(14,A,32)
870 CV=CV-1
880 IF CV=0 THEN 1070
890 GOTO 650
900 CALL SOUND(100,550,0)
910 CALL SOUND(100,450,0)
920 CALL SOUND(100,500,0)
930 CALL HCHAR(1,A,140)
940 CALL HCHAR(1,A,32)
950 IF SC=0 THEN 1030
960 IF SC=120 THEN 990
970 SC=SC+10
980 GOTO 650

```

```

990 CALL COLOR(15,8,2)
1000 CALL CHAR(144,"FF0101010101FF")
1010 SC=SC+20
1020 GOTO 980
1030 CALL COLOR(15,4,2)
1040 CALL CHAR(144,"3C7EFFFFFFF7C")
1050 SC=SC+40
1060 GOTO 980
1070 M$="GAME OVER"
1080 II=12
1090 OA=11
1100 GOSUB 1290
1110 FOR I=1 TO 1000
1120 NEXT I
1130 CALL CLEAR
1140 IF SC=HS THEN 1170
1150 INPUT "ANOTHER GAME? "ID$
1160 IF SEG$(ID$,1,1)="Y" THEN 350 ELSE 1230
1170 HS=SC
1180 PRINT "CONGRATULATIONS!!!--YOU HAVE ATTAINED THE--"HIGH SCORE.PLEASE EN
TER YOUR NAME"
1190 INPUT "NOT MORE THAN 7 LETTERS "H$
1200 IF LEN(H$)>7 THEN 1190
1210 CALL CLEAR
1220 GOTO 1150
1230 CALL CLEAR
1240 PRINT TAB(2);""BETTER LUCK NEXT TIME""
1250 FOR I=1 TO 500
1260 NEXT I
1270 CALL CLEAR
1280 END
1290 FOR N=1 TO LEN(M$)
1300 CALL HCHAR(II,N+OA,ASC(SEG$(M$,N,1)))
1310 NEXT N
1320 RETURN
1330 M$="FRUIT COCKTAIL"
1340 II=12
1350 OA=9
1360 GOSUB 1290
1370 M$="*****"
1380 II=13
1390 OA=9
1400 GOSUB 1290
1410 M$="G.W.HARTLEY"
1420 II=15
1430 OA=18
1440 GOSUB 1290
1450 FOR N=1 TO 1000
1460 NEXT N
1470 PRINT TAB(10);"INSTRUCTIONS"
1480 PRINT TAB(10);"-----"
1490 PRINT "CATCH THE FRUITS THAT ARE--"FALLING BEFORE IT SMASHES--"THE GLASS.
--"USE--"D" FOR RIGHT"
1500 PRINT "S" FOR LEFT--"THE FRUIT VALUES--"CHERRY-----10 POINTS--"ICE C
UBE-----20 POINTS"
1510 PRINT "OLIVES-----40 POINTS"
1520 PRINT "PRESS ANY KEY TO CONTINUE.."
1530 CALL KEY(0,R1,R2)
1540 IF R2=0 THEN 1530
1550 CALL CLEAR
1560 RETURN
1570 CALL CLEAR
1580 PRINT "LEVELS:""1,-----EASY""2,-----HARD""3,-----VERY HARD"
1590 INPUT "LEVEL 1-3? "D
1600 CALL CLEAR
1610 IF (D>3)&(D<1) THEN 1590
1620 IF D<1 THEN 1640
1630 LN=2
1640 IF D<2 THEN 1660
1650 LN=4
1660 IF D<3 THEN 1680
1670 LN=6
1680 RETURN

```

Micro Tip

TI-99/4A

Date your battery

The modules for the TI-99/4A contain in coded form the place and date of manufacture.

This is not an idle piece of information: the Mini Memory Module contains a battery, and that has a limited life. Although the battery is said to last for two years, that period does not start from the date you buy the module.

The life of any battery starts with the date the battery is manufactured, as some small current passes through the internal resistance of the battery. When a battery is soldered into the mini memory module, it is immediately 'on load' and running down.

The date is coded in the form week number/year, and either prefixed or suffixed with a three letter code indicating the place of manufacture.

In an early edition of (the now defunct) 'Tidings', Robin Frowd of Texas Instruments indicated that ATA was Almelo (Holland). Sorry, it isn't!

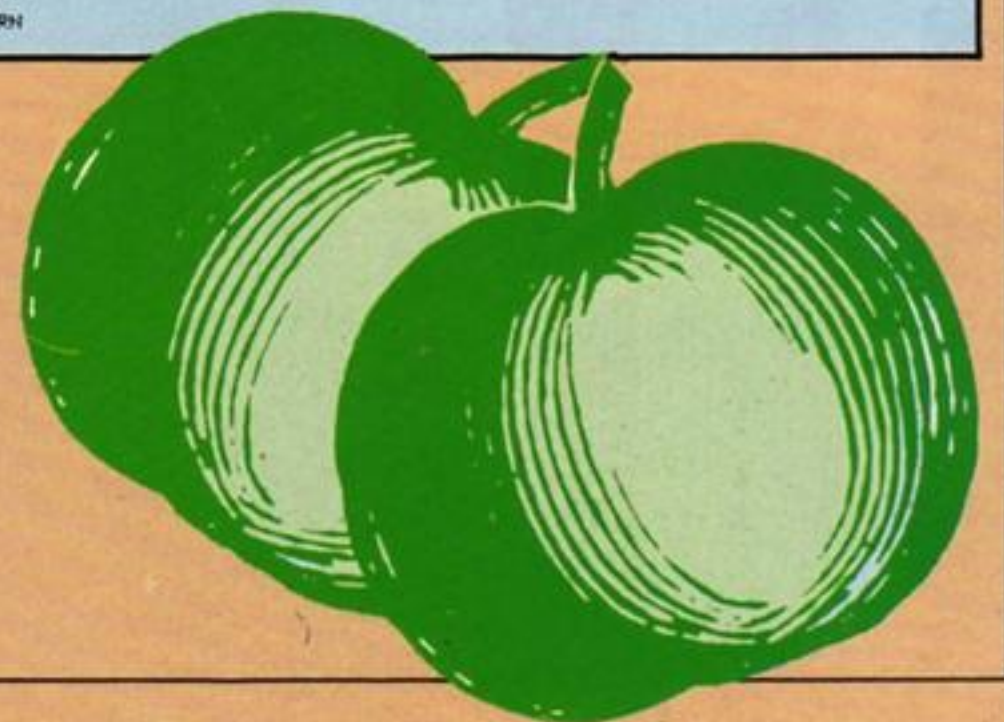
ATA appears to be AUSTIN, Texas. LTA is LUBBOCK, Texas.

HOLLAND is ACH, while ITALY is RCI.

Modules from other parties are not so encoded. I have modules from Milton Bradley (same style as TI's), Funware (similar to TI's but lacking the sliding protection plate at the back) and Romox (quite different in style: no PCB protection either, but the tracks are gold plated!).

Audiogenic have taken over the distribution of Funware modules in the UK. Watch out for them: if not cheap, they are good value for money.

Stephen Shaw



A B M Control EX BASIC, Joystick £6.95

Fantasia 99, from Christine Computing, 6 Florence Close, Watford, Herts

A B M, or Anti-Ballistic Missile Control, sets one or two players the formidable task of defending four cities and three missiles bases from destruction by bombs, bombers and cruise missiles, all in various formations.

By centring a sight on the target and pressing the fire button an A B M is launched from the player's nearest operational base. The game continues for as long as one city remains, the computer's attacks coming in successive waves.

At the end of each successful defence a score is given calculated on bonuses obtained during that round multiplied by the number of screens survived. After this score has been added to the total, the display changes colour combinations and the game continues at a more advanced level.

Although this is a very good game I was staggered at the price. In the same batch for review came this same program by Fantasia, from a different distributors which was offering it with another game for less!

J.W.

instructions	n/a
playability	80%
graphics	80%
value for money	50%



System Trader Typing Test £4.55

Solway, 6 Curzon St, Maryport, Cumbria CA15 6LL

System Trader comes complete with a Flight Manual and Trading Regulations. It's all about buying and selling commodities through

the Solar System to make a fat profit before returning to headquarters.

There's more to it than wheeling and dealing, however, as you must also take command of your spaceship, the Dragon-slayer.

On landing or take off the screen displays instrument readings for fuel, altitude, velocity and thrust. Using the

numeric keys, and spacebar in emergencies, thrust is controlled to achieve launch and, hopefully, safe landings. Correct handling of the thrust takes some getting used to — most of my descents sent me to hospital!

When you've finally worked your way round the planets you're given a comment, varying from unsuitable managerial material to Trader of the Year!

Typing Test is an entertaining program to speed up your skill at

locating letters on the keyboard. You may choose a delay factor from 0-500. After a test of 10 letters you are given a score and told the lowest highest and average delays in response. J.W.

instructions	100%
playability	70%
graphics	50%
value for money	100%



Builder/ Minefield £4.95

Lantern, 4 Haffenden Rd, Tenterden TN30 6QD

Two simple yet entertaining games which are loaded together then selected from menu. Versions are provided for both keyboard and joystick.

The object of Builder is to pick up bricks and lay them on a concrete strip. As you steer your truck round the site, puddles must be avoided as should the oily trail left behind. You may have either a static or moving truck, where moving means continually travelling in the direction last input.

In the keyboard version, however, the truck can be stopped by any key other than those

controlling direction — an oversight which allows cheating! The game ends either when all bricks have been successfully laid or if you cause so much damage that you are sacked.

In Minefield you steer a tank from the left to right avoiding mines in your path, the number of which is increased with each traverse. If this wasn't enough, a howitzer periodically fires, blocking your way even more. When the tank crashes three times the game ends, showing yards of territory gained. When this becomes too easy, you can get opt for invisible mines and then you're in trouble!

J.W.

instructions	70%
playability	75%
graphics	70%
value for money	90%



Astrowars £5

Stainless, 10 Alstone Rd, Stockport, Ches Sk4 5AH

A game which will be familiar to everyone as there is a version for just about every computer and game console.

You control a spaceship in the centre of the screen and you spin the ship, using the S and D keys, to fire at asteroids and alien spacecraft coming at you from all angles.

This cassette, however, is written in BASIC — it has to be for the standard TI — and

although it has all the ingredients of the game it is let down badly by the speed of the language.

Instructions on screen and the controls are easy to operate making the game suitable for the younger enthusiast.

Graphics and colour are used well. As already mentioned, however, the speed of the action is too slow for this sort of game and I feel therefore that the price quoted is a little high. D.B.

instructions	70%
playability	60%
graphics	70%
value for money	50%



Bob Sleigh/ Bio-rhythms £4.50

Byteware, Unit 25, Handyside Arcade, Newcastle-upon-Tyne NE1 4PZ

The object of Bob Sleigh is to steer along the track for as far as you can without crashing.

There are four courses, ranging

from "very easy, flat" to "impossible, no kids" — little does the author know that "kids" are usually best at these.

When the game is played, you discover that the different skill levels only alter the width of the track, but there are four speeds at which the course may be attempted.

The sounds and graphics were just about acceptable, but not really anything special — indeed,

I was amazed to find that one graphics character was taken straight out of the TI manual!

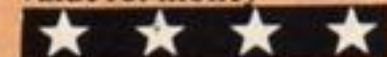
Bio-rhythms are cycles that start on the day you are born. There are three types: physical, every 23 days; psyche, every 27 days; and intellectual, every 33 days.

This program is designed to plot three graphs, as well as an average, for the next 30 days, giving an idea of how you will feel

and shape up during this time.

Even though Bob Sleigh became rather boring after a while, I thought Bio-rhythms made up for this. Good value for money. Ratings are averaged between the two. J.J.

instructions	80%
playability	70%
graphics	65%
value for money	80%



CHRISTINE COMPUTING Ltd

6 Florence Close, Watford, Herts.
(0923) 672941

TI-99/4A HARDWARE

32K Stand alone ram	£125
Arcade quality joystick	£22
Cassette interface leads – single	£5.40
– dual	£6.90
Compatible cassette recorder	£25.95
Rec + single lead	£30.95
Rec + dual lead	£31.95
Atari/Commodore to TI joystick interfaces	
single	£6
dual	£8.50

SOFTWARE

We are able to offer our own range as well as selected items from Stainless Software, Tomputer, Fantasia 99 and many more.

For full list send 50p (refund 1st order) or cheque/PO with order to above address.
Please add 50p p&p to any order under £10.

THANK YOU.

ONE MAN'S VIEW

Don't burn your TI

After two years experience with calculator programming, my introduction to computing proper came in May 1980 when I bought an NTSC TI-99/4A with a modified colour TV, for which I paid the princely sum of £1,300 (including credit charges), which I could ill afford.

I soon learned that I could not do all the things I'd wanted to, but I found this more a challenge than a disappointment. I've since sold all but my soul in order to expand my system and I've recently added a 99/4A, which is allowing me to indulge myself further in my favourite pass-time of playing detective with regard to the operating system.

I have invested a great deal in the machines, having been involved in TIHOME, the now-defunct user group, answering queries, writing for its sorely-missed newsletter, the first of several books (due out Feb. 16th), software reviews for HCW, reviewing software submissions, writing a series of mini-articles to be published soon, trying to set up a local user group, taking an occasional class in computing, trying to make time to visit owners, criticising the machines' bad points, praising the good ones, and generally boring to death anyone who stands near me long enough.



Despite TI's decision to withdraw totally from the home market (all current and forth-coming hardware, and eventually software), and despite publishers' resultant waning of interest (not that it was ever waxing, in some quarters), there is no way that I am going to 'drop' my machines. The TI-99 is, in spite of its faults, an excellent introduction, albeit only now that the price is so low, and I sincerely hope TI will not discourage third-party hardware suppliers beginning to emerge in the UK, and who could, ironically, make the 99/4A a much-sought-after machine in the future.

There are rumoured to be 200,000 – 500,000 machines still to be sold, and I have written to TI with some suggestions. Whether these are sensible/practical/acceptable remains to be seen, especially as I have since been told the outstanding consoles are almost entirely NTSC, and therefore probably unsuitable for use in Europe.

I am still relatively active, being a (back-seat) member of TIHOME's successor, the Texas Instruments Home Computer Users' Club.



The worst thing that could happen would be for grieving owners to have a small funeral pyre at the bottom of the garden. If you are such an owner, and you have yet to join an independent user group, then I would urge you to take the plunge. The longer you have your machine, the more experience you will gain, the more you're going to want to do and, most importantly, the more you're going to want to talk about it.

Magazines like Home Computing Weekly which support the TI are very rare birds, so support them with your best programs/articles/letters, etc. Keep up any contacts you may make. Whatever you do, be it buy another micro or emigrate, don't throw your TI in the loft.

If you really can't bear to have the machine in the house, donate it to a local school/youth club/any disabled person.

**Peter Brooks,
Oxford**

Letters to Peter Brooks about the TI Home Computer users' Club will be passed on.

● This space is for you to sound off about the micro scene, to present bouquets, to hurl brickbats. Share your views by sending them to Paul Liptrot, Home Computing Weekly, No. 1 Golden Square, London W1R 3AB. Please give your occupation and your interest in computing.

It's easy to complain about advertisements.

The Advertising Standards Authority. ✓
If an advertisement is wrong, we're here to put it right.
ASA Ltd, Brook House, Torrington Place, London WC1E 7HN

SOFTWARE REVIEWS

Jump Jet Dragon 32 £7.95

Dragon Data, Margam, Port Talbot, West Glamorgan SA13 2PE

It would be very easy to assume from the title that this is yet another flight simulator but not so. It is a strategic flying and shooting game where you take off in your Harrier jumpjet then increase altitude to attack enemy fighters and bombers which appear over the horizon to shoot you or your planes on the ground.

A fair amount of manoeuvring is possible with control for speed and height, though I tended to stay at one altitude and let the planes come to me, firing my cannon all the time!

This I found to be one of the

best plays especially if you could stay to the left of the screen.

The screen is well laid out and there are enough levels for a range of ages to enjoy the game, including a very simple level for children (or reviewers!). The side scroll action is a little slow but this may be due to hardware rather than software difficulties.

The only quibble I do have is that the jumpjet can accelerate off the right of the screen to reappear on the left. I would have expected the scroll routine to move faster. Exciting and quite addictive.

D.C.

instructions	70%
playability	80%
graphics	80%
value for money	80%



Naanas Spectrum £5.95

Mikro-Gen, 24 Agar Cres, Bracknell, Berks

I think the name Naanas is based on that long yellow fruit, eaten by chimpanzees. This suspicion is confirmed when the game begins.

Your pet chimp is knocking fruit down from the trees above and you are dashing about below trying to catch it in your bag, while avoiding coconuts and jumping the fallen bananas.

The skill level is variable, the graphics are good, as is the sound. In addition, there is a hall of fame feature, but I could find no way of user defining the keys as advertised on the cassette insert.

This simple game plot can cause some hoots of laughter

when played in a group.

But for solo play, I would have thought that something a little more sophisticated would be needed to sustain the interest of the dedicated games player.

Really, this game is more suitable for younger players who would appreciate the simplicity of operation and the hilarious cartoon characters, together with the wacky sound effects.

In conclusion, I would be surprised if this game found its way into the list of "arcade classics".

M.B.

instructions	80%
playability	75%
graphics	90%
value for money	75%



Spude TI-99/4A £4.95

Christine Computing, 6 Florence Close, Watford, Herts

This game is let down, as usual, by the slowness of TI BASIC, and some poorly-thought-out keyboard routines. There is also the additional irritation that even if you are successful you don't get anywhere; the machine still tells you "Bad Luck". There is no documentation and all instructions are in the program — which can be exasperating where there's a lot to remember. The aim is to go "tater-pickin'" at various seasons and to see how many points you can amass.

You are advised to start with summer, as the days are longer. As the sun sets, a shadow is cast gradually over your potato field, and it is curtains for you if it falls

on you or you walk into it. On top of this if you try to retrace your steps at all, you fall into a trench and die.

There are two kinds of potato, one worth extra points, and a timer ticks away at the bottom of the field. You can slow this timer down and gain valuable minutes. There seems little purpose to this game and players will soon tire of it.

P.B.

instructions	90%
playability	60%
graphics	75%
value for money	30%



Follow the stars for good games

Bananas, a Harrier jump-jet, potatoes and assorted insects — they are all in this batch of new games.

Beetle TI-99/4A £6

Stainless, 10 Alstone Rd, Stockport, Cheshire SK4 5AH

An original game in which you play the unusual role of a beetle.

The local gardener has carelessly stuck his spade through your nest. It is your job to collect the eggs and take them to a second nest.

You start off in a cavern with several articles of food, as well as rocks and eggs. There are also invisible spider traps which, when trodden on, cause a spider to spin a web that may trap you to cut off your escape.

As you move around you run out of energy. This is why you must eat the food. The eggs must be pushed to the next level underground. This is done by

landing on them. When you have pushed as many eggs as you want onto the next level you must crawl through a hole at the top of the screen to join them.

Although this game is enjoyable to play at first, it loses its lasting appeal quite quickly due to the slow speed of TI BASIC.

It is a great pity that the TI-99/4A is not equipped to use this original idea to the full.

J.J.

instructions	90%
playability	60%
graphics	80%
value for money	70%



Creepy Crawler Spectrum £5.95

Mikro-Gen, 24 Agar Cres, Bracknell, Berks

After it had constantly crashed on the instructions screen, I established that my Cambridge Computing Intelligent Joystick was not compatible with the program.

It was a relief to finally see the Giant Centipede rattling its way through the mushroom forest. With the offending unit unhitched, loading was error-free.

After selecting keyboard, or Mikro-Gen joystick, instructions alternated with scoring until the game is started by choosing one or two players. The player moves to

and fro, halfway up and down shooting mushrooms and the centipede, with the score kept bottom left.

A spider bounces around placing more mushrooms, helped by a bug — both attempting to eat the player. If the centipede is removed all colours change and a new centipede appears.

Graphics, colour and movement are pleasingly simple and the response to the keys very good — though no user defined keys as the insert promises. Each animal has its own sound when moving and when killed. I found it simple, pleasing and addictive!

T.W.

instructions	75%
playability	85%
graphics	65%
value for money	70%



TI-99/4A PROGRAM

Are you ready to save the world again? Texas Invaders' Revenge is a more sophisticated follow-up to my earlier program, published in HCW and called Texas Invaders (what else?).

How it works

10-20 clear screen, colour it black
30, 410-580 define characters
40-160 define colours
170-370 set up title screen
380-400 830-850 set up variables
590-660 instructions?
670-820 clear screen, position characters
860 randomise
870 position laser base
880-1070 detect key presses, randomise invader movement and mother ship firing
1080-1120 1130-1170 move laser base up, down
1180-1280 invader hit routine, reset variables for invader movement and laser range, print score
1290-1330 shorten laser range if invader moves forward, check if invader is same column as you
1340-1370 sound if invader is in same column, go to game end routine
1380-1450 mother ship fire routine
1460-1610 hit by mother ship routine, go to end routine
1620-1690 another mothership fire routine
1700 stop until "break" encountered
1710-1800 another game?
1810-1880 sound for negative answer, end
1900-2100 instructions, return to main program

It runs on the standard TI-99/4A.

You control a laser base at the extreme left of the screen and you can only move up and down.

The invader starts off on the right and moves up and down at random, slowly advancing on you.

It is under the protection of its mother ship which occasionally takes pot shots at you with one of its two powerful lasers.

It's your turn to save the world... again

Do you remember Eric Liddell's Texas Invaders program in HCW? Here's the follow-up, a more sophisticated program called (guess what?) Texas Invaders' Revenge. It runs on the unexpanded TI-99/4A

```
10 CALL CLEAR
20 CALL SCREEN(2)
30 CALL CHAR(128, "30082425599A100C")
40 CALL COLOR(1,5,2)
50 CALL COLOR(2,10,2)
60 CALL COLOR(3,6,2)
70 CALL COLOR(4,6,2)
80 FOR I=5 TO 8
90 CALL COLOR(I,14,2)
100 NEXT I
110 CALL COLOR(9,3,2)
120 CALL COLOR(10,5,2)
130 CALL COLOR(11,5,2)
140 CALL COLOR(12,16,2)
150 CALL COLOR(13,7,2)
160 CALL COLOR(16,10,2)
170 A$="          TEXAS"
180 P=4
190 GOSUB 290
200 A$="          INVADERS"
210 P=6
220 GOSUB 290
230 A$="          REVENGE"
240 P=8
250 GOSUB 290
```

```
260 FOR I=1 TO 400
270 NEXT I
280 GOTO 390
290 FOR I=1 TO LEN(A$)
300 CALL HCHAR(P,I,ASC(SEG$(A$,I,1)))
310 NEXT I
320 FOR I=1 TO 30 STEP 3
330 CALL SOUND(-1,110,I,220,I)
340 CALL HCHAR(20,I,128)
350 CALL HCHAR(20,I,32)
360 NEXT I
370 RETURN
380 SC=0
390 X=12
400 R=16
410 CALL CHAR(33, "1C3E63F8F8633E1C")
420 CALL CHAR(96, "000000FFFF")
```

```
430 CALL CHAR(97, "0000000000FF")
440 CALL CHAR(98, "0000FF")
450 CALL CHAR(104, "070F1F3D30C")
460 CALL CHAR(105, "0000C0303D1F0F07")
470 CALL CHAR(106, "C070F0F8FC7E3F1F")
480 CALL CHAR(107, "1F3F7EFCF8F0E")
490 CALL CHAR(108, "1F0F0707070F0703")
500 CALL CHAR(109, "03070F0707070F1F")
510 CALL CHAR(110, "80C0F0F1FCFFFFFF")
520 CALL CHAR(111, "FFFFFFFCF1F0C08")
530 CALL CHAR(112, "FF7F3E1C1C3E7FFF")
540 CALL CHAR(120, "0000001818")
550 CALL CHAR(121, "00003C24243C")
560 CALL CHAR(122, "007E424242427E")
570 CALL CHAR(123, "FF8181818181FF")
580 CALL CHAR(152, "F80C0E0F0F0E0CF8")
590 A$=" DO YOU WANT INSTRUCTIONSEY/NJ?"
```

Hints on conversion

TI BASIC is fairly standard with two main exceptions. Single statement lines are used and sub-routines for graphics and sound are available, prefixed with CALL, as detailed below.

CALL CLEAR clears screen

CALL SCREEN defines screen colour

CALL CHAR defines a character with a 16-digit hex string from an eight by eight grid. Other machines will use a different method

CALL COLOUR (character set, foreground, background) sets colour of character

CALL HCHAR (row, column, character code, number of repeats) places character on screen at co-ordinates specified and optionally repeats character horizontally

CALL VCHAR similar to HCHAR but repeats character vertically

CALL GCHAR (row, column, variable) similar to a screen PEEK on other machines. Gives the ASCII code at the given screen co-ordinates

CALL SOUND (D,F,V) gives sound of duration D, frequency F and volume V. Three sounds can be produced at once

If you get in its way it's tough luck, the world's doomed. But you can always play again.

Your laser has an automatic range adjuster so the invader is never out of range.

If you score a hit you get 10 points and the invader returns to its side of the screen and starts again. If you allow it to reach the same column as you then it's tough luck for you.

Keys E and X move up and down respectively and you fire with the full stop or greater than symbol keys.

Variables

SC score
R your position on screen
A,F variables for laser range
C column of invader
X row of invader

TI-99/4A PROGRAM

```

600 FOR I=1 TO LEN(A$)
610 CALL HCHAR(12,I,ASC(SEG$(A$,I,1)))
620 NEXT I
630 CALL KEY(0,K,S)
640 IF S=0 THEN 630
650 IF K=ASC("Y") THEN 1890
660 IF K=ASC("N") THEN 670 ELSE 630
670 CALL CLEAR
680 CALL HCHAR(1,8,83)
690 CALL HCHAR(1,9,67)
700 CALL HCHAR(1,10,79)
710 CALL HCHAR(1,11,82)
720 CALL HCHAR(1,12,69)
730 CALL VCHAR(2,1,152,23)
740 CALL HCHAR(10,30,104)
750 CALL HCHAR(10,31,106)
760 CALL HCHAR(11,31,108)
770 CALL HCHAR(11,32,110)
780 CALL HCHAR(16,30,105)
790 CALL HCHAR(16,31,107)
800 CALL HCHAR(15,31,109)
810 CALL HCHAR(15,32,111)
820 CALL VCHAR(12,32,112,3)
830 A=28
840 C=28
850 F=25
860 RANDOMIZE
870 CALL HCHAR(R,2,33)
880 CALL KEY(0,K,S)
890 CALL HCHAR(X,C,40)
900 CALL HCHAR(X,C,32)
910 X=INT(RND*24)+1
920 IF X=12 THEN 1290
930 IF X=13 THEN 1380
940 IF X=22 THEN 1620
950 IF X<1 THEN 910
960 CALL HCHAR(X,C,128)
970 CALL SOUND(-1,X+8000,0,7000,0)
980 IF S=0 THEN 880
990 IF K=ASC("E") THEN 1080
1000 IF K=ASC("X") THEN 1130
1010 IF K=ASC(".") THEN 1020
1020 CALL HCHAR(R,3,96,F)
1030 CALL SOUND(-10,-1,0)
1040 CALL HCHAR(R,3,32,F)
1050 CALL GCHAR(R,A,S)
1060 IF S<>32 THEN 1180
1070 GOTO 880
1080 IF R=3 THEN 880
1090 R=R-1
1100 CALL HCHAR(R,2,33)
1110 CALL HCHAR(R+1,2,32)
1120 GOTO 880
1130 IF R=23 THEN 880
1140 R=R+1
1150 CALL HCHAR(R,2,33)
1160 CALL HCHAR(R-1,2,32)
1170 GOTO 880
1180 FOR I=3 TO 16
1190 CALL COLOR(13,I,2)
1200 CALL SOUND(-200,110+I,I)
1210 NEXT I
1220 SC=SC+10
1230 CALL HCHAR(X,C,32)
1240 S$=" "&STR$(SC)
1250 FOR I=1 TO LEN(S$)
1260 CALL HCHAR(1,I,ASC(SEG$(S$,I,1)))
1270 NEXT I
1280 GOTO 830
1290 C=C-2
1300 F=F-2

```

```

1310 A=A-2
1320 IF C=2 THEN 1340
1330 GOTO 880
1340 FOR I=1100 TO 110 STEP -110
1350 CALL SOUND(-200,I,0)
1360 NEXT I
1370 GOTO 1710
1380 CALL HCHAR(10,3,97,27)
1390 CALL SOUND(-200,110,0)
1400 CALL SOUND(-200,113,0)
1410 CALL SOUND(-10,120,0)
1420 CALL GCHAR(10,2,S)
1430 CALL HCHAR(10,3,32,27)
1440 IF S<>32 THEN 1460
1450 GOTO 880
1460 FOR I=3 TO 16
1470 CALL COLOR(1,I,2)
1480 CALL SOUND(-90,900+I,0)
1490 NEXT I
1500 FOR I=120 TO 123
1510 CALL HCHAR(R,2,I)
1520 FOR D=1 TO 20
1530 NEXT D
1540 NEXT I
1550 CALL HCHAR(R,2,32)
1560 FOR I=1100 TO 110 STEP -110
1570 CALL SOUND(-200,I,0)
1580 NEXT I
1590 FOR I=1 TO 100
1600 NEXT I
1610 GOTO 1710
1620 CALL HCHAR(16,3,98,27)
1630 CALL SOUND(-200,300,0)
1640 CALL SOUND(-200,305,0)
1650 CALL SOUND(-90,310,0)
1660 CALL GCHAR(16,2,S)
1670 CALL HCHAR(16,3,32,27)
1680 IF S<>32 THEN 1460
1690 GOTO 880
1710 CALL CLEAR
1720 SC=0
1730 A$=" DO YOU WANT ANOTHER GAME(Y/N)?"
1740 FOR I=1 TO LEN(A$)
1750 CALL HCHAR(12,I,ASC(SEG$(A$,I,1)))
1760 NEXT I
1770 CALL KEY(0,K,S)

```



TI-99/4A PROGRAM

```

1780 IF S=0 THEN 1770
1790 IF K=ASC("Y") THEN 670
1800 IF K=ASC("N") THEN 1810 ELSE 1770
1810 CALL CLEAR
1820 FOR N=1 TO 3
1830 FOR I=0 TO 4
1840 CALL SOUND(-200,110,I,110+N,I)
1850 CALL SOUND(-9,110,I,220,N)
1860 NEXT I
1870 NEXT N
1880 END
1890 CALL CLEAR
1900 I$="      I N S T R U C T I O N S"
1910 P=1
1920 GOSUB 2060
1930 I$="      USE E TO MOVE UP"
1940 P=5
1950 GOSUB 2060
1960 I$="      USE X TO MOVE DOWN"
1970 P=7
1980 GOSUB 2060
1990 I$=" USE THE FULL STOP KEY TO FIRE"
2000 P=10
2010 GOSUB 2060
2020 FOR I=1 TO 200
2030 NEXT I
2040 CALL CLEAR
2050 GOTO 670
2060 FOR I=1 TO LEN(I$)
2070 CALL HCHAR(P,I,ASC(SEG$(I$,I,1)))
2080 NEXT I
2090 CALL SOUND(-1,5000,0)
2100 RETURN
    
```

Micro Tip

TI-99/4A

Secrets of the system

Here are some little-known system variables for the Atari.

POKE 82,0 gives you a full 40 column editing screen.

POKE 77,0 inside a main program loop stops the Atari going into attract mode in a joystick control program.

POKE 580,1 causes SYSTEM RESET to cold start the computer when pressed.

POKE 559,47 gives interesting results.

Before saving a program to tape, it is advisable to LPRINT beforehand, as there is a bug in the tape operating system similar to the one in early BBC operating systems.

Instead of using SETCOLOR to alter colour on the screen, altering the contents of COLOR0 to COLOR4 (708 to 712) using the formula

colour number * 16 + hue
will provide the desired results.

Mark Bullen

Micro Tip

Commodore 64

Cursor control

Owners of the Commodore 64 may find the following few POKES useful for controlling the cursor.

POKE 204, 0 before a GET will produce a flashing cursor (which is turned off with POKE 204, 1).

POKE 207, 1 before an INPUT will switch off the cursor until a key is pressed.

A repeat on all keys (not just the cursor keys and space bar) can be achieved with POKE 650, 128, POKEing this location with zero will return things to normal.

David Gristwood

MISSION - 1

BECOME A SECRET AGENT

Mission Software's complete package on Project Volcano, is probably the most advanced Adventure Computer Game Package you have considered this year. It is not just for the dedicated game player but for those that want to get involved in Secret Codes - Messages, and the seamy-side of DI6 (British Intelligence) and the K.G.B. IT IS A MUST FOR THE SINCLAIR SPECTRUM 48K AND DRAGON 32 HOME COMPUTER OWNERS.

When you receive your Adventure Game from Mission Software Limited, included in the package is your Passport to the international workings of the Secret Agents. Your personalised Passport will assist you in negotiating the K.G.B. network in the Soviet Block Countries and vital to the success of your mission. Command Headquarters will give you information that other Agents have recorded, together with their Communication Log, Mission Instructions, Secret Memos, Classified Diagram, Full Briefing Instructions and other Secret Data. IT'S UP TO YOU TO BREAK THE CODE AND BECOME A SUCCESSFUL SECRET AGENT. We rely upon our Agents to feed back information to make further projects interesting and complex, to tax your ability and to guarantee many hours of enjoyment. We believe Mission Software Limited is the only company that supports its Field Agents with a mass of Secret Information to make the Mission exciting and enjoyable which makes it a must in Computer Games.

• We know your friends will want to become Secret Agents and use your game, we will be quite happy to supply them with their Passport and Documents for £2.95 per kit. This way you always retain your personal Passport and Secret Information.

• To obtain your Game and personal Passport with all its Secret Documents send a Cheque or Postal Order for £6.95 (inc. P.P. & V.A.T.) with the Coupon at the foot of this advertisement.

• The moment you tell us you have completed your current assignment we will advise by Secret Data, details of how to penetrate one of the World's most secret establishments.

ORDER MISSION 1 TODAY AND JOIN THE GROWING NUMBER OF SECRET AGENTS OPERATING FROM COMMAND HEADQUARTERS

£6.95

Inc. P.P. & V.A.T.

MISSION SOFTWARE LIMITED

COMMAND HEADQUARTERS · 1st FLOOR
49 GRANGE ROAD · DARLINGTON · Co. DURHAM
DL1 5NB · TELEPHONE (0325) 483344

NAME

ADDRESS

HCW48

MISSION SOFTWARE LIMITED · COMMAND HEADQUARTERS · 1st FLOOR · 49 GRANGE ROAD · DARLINGTON
DURHAM · DL1 5NB · TEL. (0325) 483344

PLEASE SEND ME MISSION 1 FOR MY
(TICK AS APPLICABLE)

DRAGON 32 SINCLAIR SPECTRUM 48K

EXTRA PASSPORT CODE-BREAKER £2.95

I ENCLOSE MY TOTAL REMITTANCE OF £

CHEQUES AND POSTAL ORDERS MADE PAYABLE TO
MISSION SOFTWARE LIMITED

CVG2

Spring Heeled Jack Ex BASIC £6

Stainless, 10 Alstone Rd, Stockport, Cheshire SK4 5AH

A simple, uninvolved game, yet great fun to play and most addictive.

At the start the screen displays eight lines on each of which is a hole that glides along, at different speeds, to give the illusion of a series of moving pavements.

Your task is to guide a man from his starting position at the bottom up to the eight level by walking him along and jumping up through the holes. Should you allow him to fall or hit his head on the next level, by mis-timing his jump, a life is lost.

Control is from the keyboard, using keys of your own choice, and he may be moved left, right or up. Points are scored for reaching each level and both current and highest scores are displayed. After you have reached the top you move onto a new, more difficult screen — more holes or obstacles such as cars to avoid.

Once a life is lost you re-start at screen one, until finally all lives have been used.

J.W.

instructions	40%
playability	85%
graphics	70%
value for money	80%



Five more for your Texas

Read before you buy — a selection of games for the TI-99/4A rates by our panel

Documentation exists as the first of two programs, informing you of your task to rescue an animal supposedly at risk and apparently similar to a dragon.

Watchdogs, stun mines, an "invisible" (sic) maze, and cages are the obstacles in your path. You have five lives, but they are not really enough considering the poor response of the keyboard scans. Although the idea of giving the user instructions via a separate program is, in this field,

novel, it isn't particularly helpful as you have to make notes to use while playing. Games program written in TI BASIC are of necessity limited, and this program is no exception.

P.B.

instructions	90%
playability	40%
graphics	65%
value for money	30%



Galactic Zoo-Keeper £5.25

Microsonic, 85 Malmesbury Road, Cheadle Hulme, Cheshire

Numbering a few zoo-keepers among my friends, I was intrigued by this game, expecting it

to perhaps deal with animal conservation on a cosmic scale, or at least to elevate the general awareness of society about animals.

Not so. The game is rather disappointing, and considering the size of the listing and time to load, achieves very little, relying mainly on setting the player boring, difficult, and unrewarding tasks.

Hunchback Havock £5.95

Lantern, 4 Haffenden Rd, Tenterden, Kent

A very attractive presentation in which you collect jewels hidden in a nuclear powered castle. Guide Egor through the castle unlocking doors and tackling fires or other hazards in order to turn on the reactor which makes the jewels visible.

There is a very short time limit before the reactor explodes ending the game. The main difficulty is Egor cannot carry both key and the fire extinguisher so he must scuttle back and forth.

Four screens show different parts of the castle. Hazards come in six forms, including electrical faults and gas leaks, dealt with in a similar fashion to fires. This

does not add more difficulty, but introduces some graphic variety. Since time is very limited I found it impossible to progress beyond screen four, so I cheated by increasing the time limit so as to view all 24 screens — the same four repeated for six types of hazards. Play might become repetitive if you get that far — not a problem for me though!

I would like to see a time count-down marker on screen during play so as to monitor progress. Otherwise, a well designed game with excellent graphics.

C.E.

instructions	70%
playability	90%
graphics	100%
value for money	90%



Caveman EX BASIC, Joystick £4.95

Warwick, 40 Kingsway, Newby, Scarborough, N. Yorks YO12 6SG

Your objective is to recover the throne of the Inkaplunka king. To reach this treasure you must descend through five tunnels and brave the different hazards on each level. These include moving mines, flying shells, dangerous ground, hovering discs and boulders. Using the joystick in various ways according to which level you are on, the throne must be reached in the best possible time.

The screen display shows all five levels, the time, best time and number of lives remaining — from a maximum of five. Access

to each level is obtained by descending a ladder.

Although instructions are provided on screen, because they are loaded separately you need to make notes of how to use the joystick on each level before actually starting. Written instructions on the cassette insert would be more helpful.

Any enjoyment of the game itself is spoilt by various bugs inherent in the program. I failed to get any further than level three, being prevented not only by bad luck but by two BAD VALUES and a SYNTAX ERROR.

J.W.

instructions	80%
playability	50%
graphics	65%
value for money	50%



Bouncing Bugs £6

Stainless, 10 Alstone Rd, Stockport, Cheshire SK4 5AH

Three games in one, each set on a different screen but the style of play remains the same for all.

The stars are five differently coloured bugs. Each in turn

stands on a springboard positioned to the right of the screen. Using keyboard control the springboard must be raised or lowered to the correct height from which to launch the bug, so he land on various targets to the left.

Once released the bug bounces off any walls that he contacts, so keys 1-9 are also used to determine the strength of his bounce.

Each game may be played by up

to 10 people who may compete in one to 10 rounds. Points are awarded for every successful landing, the total being displayed after each completed round.

Since there is no random element, before long the exact requirements to land the bugs can be remembered. In the multi-player situation the last player undoubtedly has an unfair advantage as he is able to benefit

from all previous player's experiments and successes.

J.W.

instructions	90%
playability	50%
graphics	60%
value for money	60%



The object of Texas Telephone is to connect together two telephones through four switches — it's not as easy as it sounds.

The switches are "level" switches as found in telephone exchanges. The computer sounds a series of "pulses" corresponding to the digit dialled. The switch then steps down the "levels". You have to stop it on the level you think has been sounded.

Hints on conversion

TI BASIC is fairly standard with two main exceptions. Single statement lines are used and sub-routines for graphics and sound are available, prefixed with CALL, as detailed below.

CALL CLEAR clears screen

CALL SCREEN defines a character with a 16-digit hex string from an eight by eight grid. Other machines will use a different method

CALL COLOUR (character set, foreground, background) sets colour of character

CALL HCHAR (row, column, character code, number of repeats) places character on screen at co-ordinates specified and optionally repeats character horizontally

CALL VCHAR similar to HCHAR but repeats character vertically

CALL GCHAR (row, column, variable) similar to a screen PEEK on other machines. Gives the ASCII code at the given screen co-ordinates

CALL SOUND (D,F,V) gives sound of duration D, frequency F and volume V. Three sounds can be produced at once

If you are correct you move onto the next switch and finally ring the distant telephone. If you guess wrongly you get the "number unobtainable" tone, and a try again instruction.

Dial in to the Texas Telephones

You thought making a call was as easy as picking up the phone and dialling? P. G. Eyre, who wrote this program for the standard TI-99/4A, has got news for you...

How it works

30-795 main program
800-840 generates dial pulses sound
900-980 routine to step switch
1000-1020 simple time delay
1200-1295 set playing field
1800-1895 generates easier dial pulses sound
1900-1995 generates harder dial pulses sound
2000-2100 defines screen characters
3000-3190 introduction
3500-3640 ringing sound
4000-4195 playing instructions
5000-5070 failure routine

The computer gives playing instructions throughout the game. There are two levels of play: key 0 is easy, key 1 is harder. The only other key used is the S key, to start the game and stop the stepping of the switch.

The program makes use of many sub-routines. These make it easier to understand how it runs. There are many REM statements. As there are no jumps to any REMs, these may be omitted during typing in the program on the computer.

Variables

AS input variable to start game
D level of difficulty input variable
E,F,G data reading variable
K sets screen position of level numbers on switch
L,M,X,Y positions of switch pointer
P value of random number, generated for dialled digits
T timing variable
X,Y print positions
Z timing delay variable, also controls output of dial pulses sound and controls X position of characters during introduction

```
30 GOSUB 2000
40 GOSUB 3000
99 REM PLAY INSTRUCTIONS
100 GOSUB 4000
105 CALL SCREEN(16)
110 PRINT "PRESS KEY S TO START."
115 PRINT "THEN PRESS enter."
120 INPUT AS
130 IF AS<>"S" THEN 140 ELSE 200
140 GOSUB 4000
150 GOTO 110
200 CALL CLEAR
210 PRINT "PRESS 0 FOR EASY GAME."
220 PRINT ""
230 PRINT "PRESS 1 FOR HARDER GAME."
240 PRINT ""
245 PRINT "PRESS 0 OR 1."
246 PRINT ""
250 PRINT "THEN PRESS enter KEY."
251 FOR Z=1 TO 6
252 PRINT " "
253 NEXT Z
255 INPUT D
260 IF D>1 THEN 200 ELSE 270
265 REM SET PLAY FIELD
270 CALL CLEAR
272 CALL SCREEN(16)
275 K=9
280 GOSUB 1200
290 FOR Z=1 TO 400
292 NEXT Z
299 REM SOUND DIAL PULSES
300 GOSUB 800
339 REM STEP SWITCH
340 L=10
350 M=11
360 GOSUB 900
399 REM DRAW NEXT SWITCH
400 K=13
410 GOSUB 1200
420 FOR Z=1 TO 400
430 NEXT Z
```

```
439 REM 2nd SOUND OF DIAL PULSES
440 GOSUB 800
449 REM 2nd SWITCH STEP
450 L=10
460 M=15
470 GOSUB 900
479 REM DRAW 3rd SWITCH
480 K=17
490 GOSUB 1200
500 FOR Z=1 TO 400
510 NEXT Z
519 REM 3rd SOUND OF DIAL PULSES
520 GOSUB 800
529 REM 3rd STEP OF SWITCH
530 L=10
540 M=19
550 GOSUB 900
559 REM DRAW 4th SWITCH
560 K=21
570 GOSUB 1200
580 FOR Z=1 TO 400
590 NEXT Z
599 REM 4th SOUND OF DIAL PULSES
600 GOSUB 800
609 REM 4th SWITCH STEP
610 L=10
615 M=23
620 GOSUB 900
629 REM ALL OK
630 GOSUB 3500
632 CALL COLOR(15,14,16)
635 CALL HCHAR(12,27,144)
640 GOSUB 3500
642 GOSUB 3500
645 CALL HCHAR(12,27,145)
650 CALL HCHAR(11,26,146)
660 FOR Z=1 TO 400
670 NEXT Z
680 PRINT "PRESS KEY S FOR NEW GAME."
685 PRINT "THEN PRESS enter."
690 INPUT AS
700 IF AS<>"S" THEN 680 ELSE 200
```

```
795 END
799 REM SOUND DIAL PULSES
800 IF D=0 THEN 810 ELSE 820
810 GOSUB 1000
820 IF D=1 THEN 830 ELSE 840
830 GOSUB 1900
840 RETURN
899 REM STEP SWITCH
900 Y=L
905 X=M
910 CALL HCHAR(Y,X,133)
915 Y=Y+1
920 IF Y=20 THEN 5000 ELSE 925
925 CALL HCHAR(Y,X,132)
930 FOR Z=1 TO 10
935 NEXT Z
940 CALL KEY(0,KEY,STATUS)
945 IF STATUS=0 THEN 910 ELSE 950
950 IF KEY=03 THEN 955 ELSE 910
955 IF Y=P+10 THEN 900 ELSE 5000
960 IF Y=20 THEN 5000
980 RETURN
1000 FOR Z=1 TO 2000
1010 NEXT Z
1020 RETURN
1199 REM SET PLAY FIELD
1200 CALL COLOR(13,2,16)
1201 CALL HCHAR(12,4,120)
1202 CALL COLOR(14,5,16)
1210 FOR Z=1 TO 150
1215 NEXT Z
1220 CALL HCHAR(12,4,129)
1225 CALL HCHAR(11,5,130)
1230 FOR Y=11 TO 19
1235 X=K
1240 CALL HCHAR(Y,X,30+Y)
1242 CALL HCHAR(Y,X+1,136)
1245 NEXT Y
1247 Y=11
1250 CALL HCHAR(Y-1,X+2,132)
1255 RETURN
1799 REM DIAL PULSES SOUND
1800 RANDOMIZE
1810 P=INT(9*RAND)*1
1820 FOR Z=1 TO P
```

```
1830 CALL SOUND(33,110,2)
1840 FOR T=1 TO 30
1850 NEXT T
1860 NEXT Z
1895 RETURN
1900 REM HARDER PULSING
1900 RANDOMIZE
1910 P=INT(9*RAND)*1
1920 FOR Z=1 TO P
1930 CALL SOUND(33,110,2)
1940 FOR T=1 TO 5
1950 NEXT T
1960 NEXT Z
1995 RETURN
1999 REM TELE REC ON
2000 CALL CHAR(120,"7EFFDB103C66E7FF")
2019 REM TELE REC OFF
2020 CALL CHAR(125,"0101191H3C66E7FF")
2030 CALL CHAR(130,"7EFFDB1010204F0")
2039 REM (LEVEL) CHAR
2040 CALL CHAR(136,"000000FEFE000000")
2049 REM POINTER
2050 CALL CHAR(132,"00002060E0602000")
2059 REM BLANK SPACE
2060 CALL CHAR(133,"0000000000000000")
2069 REM END OF GAME TELE
2070 CALL CHAR(144,"7EFFDB103C66E7FF")
2080 CALL CHAR(145,"0101191H3C66E7FF")
2090 CALL CHAR(146,"7EFFDB1010204F0")
2100 RETURN
2999 REM INTRODUCTION
3000 CALL CLEAR
3010 CALL SCREEN(12)
3020 GOSUB 3500
3040 GOSUB 3500
3050 CALL COLOR(13,2,12)
3055 FOR Z=4 TO 20 STEP 2
3060 CALL HCHAR(6,Z,120)
3065 NEXT Z
3066 GOSUB 3500
3068 FOR Z=4 TO 20 STEP 2
3070 CALL HCHAR(19,Z,120)
3072 NEXT Z
3075 GOSUB 3500
3078 FOR Z=6 TO 16 STEP 2
3080 CALL VCHAR(Z,4,120)
```

```
3084 NEXT Z
3090 GOSUB 3500
3095 FOR Z=6 TO 16 STEP 2
3100 CALL VCHAR(Z,20,120)
3101 NEXT Z
3102 GOSUB 3500
3103 FOR Z=4 TO 20 STEP 2
3105 CALL HCHAR(6,Z,129)
3110 CALL HCHAR(5,Z+1,130)
3112 NEXT Z
3120 FOR Z=9 TO 12
3125 CALL COLOR(Z,2,12)
3130 NEXT Z
3135 RESTORE 3600
3140 FOR Z=1 TO 22
3145 READ E,F,G
3150 CALL HCHAR(E,F,G)
3155 NEXT Z
3160 FOR Z=1 TO 1000
3165 NEXT Z
3190 RETURN
3499 REM RINGING
3500 CALL SOUND(300,210,4,320,4)
3510 FOR Z=1 TO 130
3520 NEXT Z
3530 CALL SOUND(300,210,4,320,4)
3540 FOR Z=1 TO 400
3550 NEXT Z
3560 RETURN
3599 REM DATA FOR INTRO.
3600 DATA 0,11,116,0,12,101,0,13,120,0,14,97,0,15,115
3610 DATA 10,10,116,10,11,101,10,12,100,10,13,101,10,14,112,10,15,104,10,16,111
3620 DATA 10,17,110,10,18,101,10,19,115
3630 DATA 12,13,90,12,14,121
3640 DATA 14,10,99,14,12,101,14,13,121,14,14,114,14,15,101
3999 REM PLAY INSTRUCTIONS
4000 CALL CLEAR
4001 CALL SCREEN(15)
4002 FOR Z=9 TO 12
4004 CALL COLOR(15,13,15)
4005 NEXT Z
4010 PRINT "connect the telephones."
4020 PRINT " "
4030 PRINT "you have two difficulty "
4040 PRINT "levels."
4050 PRINT " "
4060 PRINT "after hearing dial pulses."
4070 PRINT "the switch will step."
```

```
4075 PRINT " "
4080 PRINT "Press key s to stop."
4090 PRINT "on the number dialled."
4100 PRINT " "
4110 PRINT "you have four switches to "
4120 PRINT "connect the call."
4130 PRINT " "
4195 RETURN
4999 REM FAIL ROUTINE
5000 CALL CLEAR
5010 CALL SCREEN(10)
5030 CALL SOUND(3000,400,4)
5040 PRINT TAB(6);"WRONG NUMBER"
5050 PRINT "PRESS S KEY FOR NEW GAME."
5055 PRINT "THEN PRESS enter KEY."
5058 FOR Z=1 TO 8
5060 PRINT " "
5062 NEXT Z
5065 INPUT AS
5070 IF AS="S" THEN 200 ELSE 5000
```


Minnesota schools get \$ millions for software

California isn't the only technologically aware state in the country, as far as education goes. In a recent move, the Minnesota legislature passed a \$5.8 million Technology and Educational Improvement Act to basically force school districts to make the best possible use of new technology.

One goal of the act is to allow school districts to purchase high quality software and to have schools develop plans to use it, as well as train staff in teaching and implementing high-tech applications. The act requires each of the state's school districts to develop what they call a technology utilisation plan. The plan must be developed in co-operation with local advisory committees comprised of community members, staff, and parents. No district will receive less than \$500 to aid in the creation of their local plan.

Each plan must meet seven strict criteria in order to be acceptable. Of the seven factors, the one I can best appreciate requires the local district to indicate how their plan to implement the new technology will be applied to people of all ages, including women, minorities, and the disabled. All the criteria are intended to gain the co-operation and participation of the local communities.

What is equally interesting is that this is a software funding program only. How the districts attain the computers and peripherals is left up to them. Because of the manufacturers' donation plans, obtaining the hardware is not seen as much of a problem. It is interesting that many local programmers now have a chance to get exposure for their efforts for very little in the way of advertising fees.

In fact, the procedure for getting software approved for sale to the Minnesota school districts under this program is quite simple. A disc along with any other material (like documentation, for instance) should be sent to Dr Gilbert Valdez, Minnesota Department of Education, Capitol Square Building, 550 Cedar, St. Paul, Minnesota 55101. If you are interested in the scheme and want to question Dr Valdez directly you can call him on (612) 296-4067.

I hope this scheme serves as an example of what can be done to promote computer education in school systems. It is certainly clear that many systems will resist change until, like many districts in Minnesota, it is literally forced upon them. It is also clear that, in this respect at least, we still have much to learn from our British cousins.



As promised, here is some good news for TI-99/4A owners. There are quite a few vendors who plan to continue to market support products for the discontinued machine. It is sort of like a good news — bad news joke, though.

The good news is that there will be support for some time to come from a variety of sources; the bad news is that it will never be as good as when the machine was in current production — but that is usually the case when any product is discontinued. Never mind. There is still an installed user base of approximately two million people over here which is not exactly an insignificant number, so support will certainly not disappear overnight. Here are some of the more interesting products I know about for the little box.

First of all, a firm called CorComp of Laguna Hills, California, (714) 855-8033 is now developing a microcomputer that will be compatible with all TI software. There is at present no name for the new unit, but it is expected out soon. It will have 64K of RAM, a built-in speech synthesizer, and built-in BASIC. It is supposed to sell for less than \$500 and have an optional disc drive. CorComp is not without previous experience with the TI. It presently makes 32K RAM boards and RS-232 boards for the TI expansion box. CorComp is also planning to make a smaller version of the TI expansion box (no disc drive) that will accept the various boards made for the expansion box.

If you want to buy a disc drive unit for your TI, from someone other than TI, you are limited to the product made by Percom Corporation, of Dallas, Texas. Percom makes drives, and good ones at that, for just about every home computer, so this is no fly-

by-night organisation. Their TX 99-S1 drive carried a suggested list of \$449 which includes the controller card. Percom has about 600 of these drives on hand worldwide at the present time.

It also makes a second drive for the TI called the TX 99-A1 which lists for \$399. At the time of writing, it isn't clear if Percom will continue to manufacture these units. It all hinges on demand.

A company called Myarc of Basking Ridge, New Jersey, is planning to continue to manufacture its line of 5-, 10-, and 15-megabyte hard disc drives for the TI. This company has been making these units since earlier this year and apparently is selling many of them in South America where the TI is used mostly as a business computer.

Any number of small companies are planning to continue making cards and accessories to plug into the expansion box. These firms include A/D Electronics, of Sacramento, and Doryt Systems of Glen Cove, New York. Newport Controls, of Bishop, California, still plans to market its \$30 joystick. 99er Home Computer Magazine of Eugene Oregon is still going strong, and has published a directory of third party hardware products in its September issue.

Of greater concern than hardware is the software issue. While many manufacturers are foregoing offering their hardware products in retail outlets in favour of doing so by mail order, the software vendors are unsure of which way to go.

TI plans to sell off all its remaining software. The publisher of 99er feels that this will allow the third party software market to really take off. Sierra On-Line Software is not so sure. It took them months to complete successful negotiations that allowed TI to manufacture and distribute game cartridges for which Sierra would receive royalties. Of the six games TI planned to produce and sell in the manner, two are already being produced (these being Jawbreaker and Crossfire) and are therefore likely to be released, although who know for how long.

TI has not said anything yet about releasing the other four, and Sierra tends to think that if TI sends the games back that it will not market the games either. It seems clear from present indications that most retail outlets would rather use their shelf space for software other than that designed to run on the TI.

One possible reason for this is the system developed by Romox and mentioned in this column a while back. Using the Romox system, retailers can download software into reusable cartridges getting the programs over the phone lines, thus decreasing stock requirements.

Another problem facing the software manufacturers is that most of them are not set up to sell directly to the public, so if they decide to go the mail order route, they would have to restructure their organisation. This could be risky as no one knows how to predict what sales will be as a result. In order to do an effective mail order business, a firm needs a good direct mailing list. The best source of such a list is of course Texas Instruments itself, which has a record of almost every owner of a TI-99/4A. Yet so far TI has not offered this list to anyone at any price, not even the main TI user group, International 99 Users Group, of Bethany, Oklahoma.

One thing is for sure — I'll be willing to bet money that Atari is very upset over TI's rapid departure from the field. Rumour has it that Atari has already produced about a quarter of a million game cartridges for the TI under its new software company, Atarisoft Inc. A loss in that line now would be the last thing that Atari needs after its problems of the last year or so. Oh well, no one said this business was easy.

Hill Street Blues fans will appreciate this warning to potential software writers: Let's be careful out there. See you next week.

Out of room again — see you next week.

Bud Izen
Fairfield, California

LETTERS

Charting the scores

Why not have a chart showing readers' highest scores on games? Choose a game from each of your software charts, notify the readers in advance which it is, and get them to send in a photograph of their scores. At the end of the year you could compile a chart showing the highest scores. For each game you could set a target that readers have to beat before sending in their own scores.

Ian Reichard, Sunderland

A good idea, but with some practical difficulties. For example it's not unknown for computer users to break into programs. They could fake a high score for a picture.

What do you think? Let's have your letters.

Seg's counsel over Texas

At the risk of initiating an on-going correspondence situation — as our cousins across the Atlantic might say — I feel obliged to reply to criticisms made about me by William Smith in HCW 46.

He seems to imply that I am disappointed with my TI-99/4A. Far from it. It is precisely because it is such a good computer that I am unhappy; for Texas Instruments to axe the 99 may be commercially and financially expedient, but what a waste of potential! Hang your heads in shame, TI.

Perhaps I was unfair to TI owners; I know any true 99er will defend the computer to the death against the hordes of unwashed Spectrum users and the like. On the other hand, there are quite a few Texas users who, on discovering that TI BASIC will not send aliens zooming across the screen at the speed of light, banish their machines to the attic prematurely.

Now, I use my 99 almost exclusively for programming. In fact, I haven't bought on single item of software. Before you scream "Apathy!" let me explain. Being a member of the three million jobless club, I don't have a great deal of money left over at the end of the fortnight, and what little I have goes towards the expansion of my system. In fact, I've already saved absolutely nothing!

As for the computing press (not HCW), there can be no defence here.

Send your letter to Letters, Home Computing Weekly, No.1 Golden Square, London W1R 3AB. Don't forget to name your computer — the best letter could win £5-worth of software. Got a problem with your micro? We'll soon be starting a queries page, so send your technical questions to Queries at the above address



For example: I recently managed to acquire Extended BASIC and a speech unit by virtue of a small windfall and I am delighted with both. However, this set me thinking. Have you ever seen a review of X-BASIC in the computing press? Did you know there are in fact two home computers with named procedures (counting BBC and Electron as one)?

I also recently read a review of speech add-ons for home micros in a computing rag which shall remain nameless. Was the best speech synthesis unit for any computer even mentioned? Damn well right it wasn't!

I rest my case, m'lud.
Seg Gorham, Wakefield, West Yorks

We have received a lot of letters about Texas Instruments ending the production of the 99/4a and we are printing a small selection of them. However, most say roughly the same thing — how disappointed TI owners are and what a fine machine it really is. Please now write only if you have something genuinely new to say. And, as with all letters for publication, keep it brief!

TI: I'm not changing

Hear, hear, to Seg Gorham's letter in HCW 42. I, too, am a proud owner of a TI-99/4A and am disturbed by recent developments. It is now possible to buy a TI-99/4A for almost half the cost I paid for mine. This fact alone should make it popular with a bargain-hungry public if it weren't for the apathy and scant support it receives.

However, is it just coincidence or is it true that software (especially games) for the TI has become more prevalent in recent months? It would appear so.

Owners can only hope that those magazines (such

as HCW) and the few friendly dealers and software houses that do support the TI continue to do so for a long time to come, as I for one am in no hurry to change to a five function button Spectrum.

No doubt this argument will continue for some time but if you print this letter I would be fascinated to hear the response, not from other TI users caught in this situation but also retailers and software creators.

Stuart Gilmour, Glasgow

Zap-free games

Regarding the letter (HCW 41) entitled: "Peace on earth — and elsewhere", I would first like to point out that there are lots of other games available for all machines that do not involve zapping aliens, e.g. Pacman or Donkey Kong.

Secondly, if someone feels that he must go out and kill someone, and someone plonks a game of Space Invaders in his hand, he will probably sit down and release his anger or hatred on the game.

Simon Drabble, Caterham, Surrey

Bring back our TI

I am writing in response to Seg Gorham's letter in HCW 42 concerning the withdrawal from the market of the TI-99/4A.

My husband and I bought our TI in October from a large nation-wide department store — which shall remain unnamed, much to my regret — because we thought it would be a good investment. However, we had no idea of what TI intended to do. I can now see why the salesman was so eager for us to buy the TI-99/4A: to clear their stock.

I have barely started learning the BASIC statements etc, let alone doing my own programming, so the chances of any peripherals being around when I'm capable of using them is negligible.

I would like to ask this of all shops, magazines and anyone in the home computer market: please support the TI and no more cold-shouldering. And, TI, please bring our computer back.

Thank you, HCW, for supporting us angry and disappointed TI owners.

Mrs S. J. Harris, Winchwen, Swansea

TI-99/4A PROGRAM

How it works

100 REM
110-160 Y or N to instructions
170-290 instructions
300-830 define characters for screen, laser beam and set screen
840 set random column for alien descent
850-920 define alien, move alien
930-950 limit quantity of aliens
960-1090 fire laser beam
1100-1120 limit quantity of laser shots
1130-1140 stop alien when laser is fired
1150 decide if laser is on target
1160-1240 laser miss routine
1250-1270 score generator
1280-1430 game over
1440-1680 explosion routine

Laser Defence seems simple to play — there's just one key to press — but it's not as easy as it looks.

Use any key, I recommend the space bar, to fire your laser beam at the aliens as they ascend.

Only one alien will appear at a time and your number of shots is not restricted, although you can be attacked by up to 20 aliens and then you have a maximum of 20 shots.

Variables

E random position of alien between columns 1 and 24

C row number of alien in steps of one from row 1 to 21

I column number of laser gun in steps of -1 from column 29 to 1

The other variables explain themselves

Don't be fooled... it's not so easy

Here they come again. Those aliens are determined to conquer us. You can stop them. Quickly type in Tim Freeman's program for the standard TI-99/4A

Hints on conversion

TI BASIC is fairly standard with two main exceptions. Single statement lines are used and sub-routines for graphics and sound are available, prefixed with CALL, as detailed below.

CALL CLEAR clears screen

CALL SCREEN defines screen colour

CALL CHAR defines a character with a 16-digit hex string from an eight by eight grid. Other machines will use a different method

CALL COLOUR (character set, foreground, background) sets colour of character

CALL HCHAR (row, column, character code, number of repeats) places character on screen at co-ordinates specified and optionally repeats character horizontally

CALL VCHAR similar to HCHAR but repeats character vertically

CALL GCHAR (row, column, variable) similar to a screen PEEK on other machines. Gives the ASCII code at the given screen co-ordinates

CALL SOUND (D,F,V) gives sound of duration D, frequency F and volume V. Three sounds can be produced at once

It is wise to aim carefully and hit each alien in turn.

The best score possible is zero, which means that all aliens were destroyed without any harm to the city or its people.

```
100 REM LASER DEFENCE BY T FREEMAN
110 CALL CLEAR
120 PRINT "DO YOU WANT INSTRUCTIONS?"
130 CALL KEY(0,K,S)
140 IF S=0 THEN 130
150 IF K=78 THEN 300
160 IF K=89 THEN 170 ELSE 130
170 CALL CLEAR
180 PRINT "YOU ARE IN THE LASER CONTROLTOWER AND YOU ARE THE CITY'S ONLY FORM OF DEFENCE-----"
190 PRINT "      * you will be attacked by upto 20 aliens"
200 PRINT "      * you can shoot up to 20 times *"
210 PRINT "*****"
220 PRINT
230 PRINT "YOUR OBJECTIVE IS TO DESTROY ALL ALIENS...YOUR SCORE WILL DEPEND ON THIS"
240 PRINT
250 PRINT "you should try for a score that is as close to zero as possible...the higher the score the worse your aim is"
260 PRINT "      use the space bar to fire"
270 PRINT "      PRESS ANY KEY TO BEGIN"
280 CALL KEY(0,K,S)
290 IF S=0 THEN 280 ELSE 300
300 RANDOMIZE
310 CALL SCREEN(0)
320 CALL CLEAR
330 CALL COLOR(1,13,2)
340 CALL CHAR(33,"FFFFFFFFFFFFFF")
350 CALL COLOR(2,7,2)
360 CALL CHAR(40,"FFFFFFFFFFFFFF")
370 CALL CHAR(41,"7C0603010F1F3F7F")
380 CALL CHAR(42,"3E6C080F0F8FCFE")
```

```
1410 PRINT "GAME OVER"
1420 PRINT "      GAME OVER"
1430 END
1440 CALL SOUND(-10,110,0,330,1,1000,0,-3,0)
1450 CALL HCHAR(C,1,36)
1460 CALL SOUND(-9,120,0,340,1,1010,0,-3,0)
1470 CALL HCHAR(C,1,40)
1480 CALL SOUND(-8,130,0,350,1,1020,0,-3,0)
1490 CALL HCHAR(C,1,37)
1500 CALL SOUND(-7,140,0,360,1,1030,0,-3,0)
1510 CALL HCHAR(C,1,40)
1520 CALL SOUND(-6,150,0,370,1,1040,0,-3,0)
1530 CALL HCHAR(C,1,36)
1540 CALL SOUND(-5,160,0,380,1,1050,0,-3,0)
1550 CALL HCHAR(C,1,40)
1560 CALL SOUND(-4,170,0,390,1,1060,0,-3,0)
1570 CALL HCHAR(C,1,37)
1580 CALL SOUND(-3,180,0,390,1,1070,0,-3,0)
1590 CALL HCHAR(C,1,40)
1600 CALL SOUND(-2,190,0,400,1,1080,0,-3,0)
1610 CALL HCHAR(C,1,36)
1620 CALL SOUND(-1,200,0,410,1,1090,0,-3,0)
1630 CALL HCHAR(C,1,40)
1640 CALL SOUND(1,210,0,420,1,1100,0,-3,0)
1650 CALL HCHAR(C,1,104)
1660 CALL SOUND(2,220,0,430,1,1110,0,-3,0)
1670 IF HITS=20 THEN 32767 ELSE 840
1680 END
```

```
1020 CALL HCHAR(C,E,36)
1030 AB="0002093CFF3C0002"
1040 CALL SOUND(3750,110,0,500,0,-8,0)
1050 CALL CHAR(120,AB)
1060 CALL CHAR(104,0#)
1070 CALL COLOR(12,4,2)
1080 CALL COLOR(10,2,2)
1090 FOR I=29 TO 1 STEP -1
1100 IF I=1 THEN 1110 ELSE 1130
1110 SHOTS=SHOTS+1
1120 IF SHOTS=20 THEN 1270 ELSE 1130
1130 CALL HCHAR(12,I,120)
1140 CALL HCHAR(12,I,104)
1150 IF (12/I)*(12/I)<(C/E)<(C/E)<(C/E) THEN 1250
1160 NEXT I
1170 CALL HCHAR(C,E,104)
1180 GOTO 1080
1190 IF C=21 THEN 840
1200 IF C=20 THEN 1210 ELSE 1220
1210 CALL SOUND(200,110,0,500,0,250,0,-7,0)
1220 E=E
1230 NEXT C
1240 GOTO 920
1250 HITS=HITS+1
1260 IF HITS=20 THEN 1450 ELSE 1270
1270 SCORE=(ALIENS+SHOTS)-(HITS+HITS)*10
1280 CALL CLEAR
1290 CALL SCREEN(4)
1300 CALL COLOR(4,2,4)
1310 CALL COLOR(1,4,4)
1320 PRINT "YOUR MISSION WAS ACHIEVED AFTER YOU ALLOWED THE ALIEN TO KILL "S"
1330 PRINT "THOUSAND PEOPLE IN YOUR CITY-----"
1340 PRINT "      DO YOU WANT TO PLAY AGAIN?"
1350 PRINT
1360 CALL KEY(0,K,S)
1370 IF S=0 THEN 1360
1380 IF K=78 THEN 300
1390 IF K=89 THEN 1400 ELSE 1360
1400 CALL CLEAR
```

```
390 CALL CHAR(43,"FFFFFFFFF80C060")
400 CALL CHAR(44,"FFFFFFFFF010306")
410 CALL CHAR(45,"1F3061D99DA6938B")
420 CALL CHAR(46,"F80CC6C3C187EFE7")
430 CALL CHAR(60,"C07038383870C0")
440 CALL COLOR(15,3,2)
450 CALL CHAR(144,"8040201010205898")
460 CALL CHAR(145,"1010101010101010")
470 CALL COLOR(4,11,2)
480 CALL CHAR(61,"91523C3FFC3C4A89")
490 CALL COLOR(16,14,2)
500 CALL CHAR(152,"7E7E5A7EFFFF8BEF")
510 CALL HCHAR(20,2,152)
520 CALL HCHAR(20,5,152)
530 CALL HCHAR(20,7,152,2)
540 CALL HCHAR(20,12,152)
550 CALL HCHAR(12,30,144)
560 CALL VCHAR(13,30,145,4)
570 CALL HCHAR(20,18,152,2)
580 CALL HCHAR(20,24,152)
590 CALL HCHAR(20,3,152)
600 CALL HCHAR(20,9,152,2)
610 CALL HCHAR(20,14,152,3)
620 CALL HCHAR(20,19,152,5)
630 CALL HCHAR(3,5,61)
640 CALL HCHAR(9,23,61)
650 CALL HCHAR(3,16,61)
660 CALL HCHAR(4,29,61)
670 CALL HCHAR(7,8,61)
680 CALL HCHAR(10,32,61)
690 CALL HCHAR(6,19,61)
700 CALL HCHAR(2,12,61)
710 CALL HCHAR(5,26,61)
720 CALL HCHAR(21,1,33,128)
730 CALL HCHAR(17,25,40,6)
740 CALL HCHAR(18,26,40,4)
750 CALL HCHAR(19,27,40,2)
760 CALL HCHAR(20,26,40,7)
770 CALL HCHAR(16,27,41)
780 CALL HCHAR(16,28,42)
790 CALL HCHAR(15,27,43)
800 CALL HCHAR(15,28,44)
810 CALL HCHAR(14,27,45)
820 CALL HCHAR(14,28,46)
830 CALL HCHAR(4,23,60)
840 E=INT(23*RND)+1
850 C="C3241010101024C3"
860 E="0000FF1010FF0000"
870 D="FFFFFFFFFFFFFF"
880 CALL CHAR(36,C)
890 CALL CHAR(37,E)
900 CALL CHAR(104,D)
910 CALL COLOR(10,2,2)
920 FOR C=1 TO 21
930 IF C=1 THEN 940 ELSE 960
940 ALIEN=ALIEN+1
950 IF ALIEN=20 THEN 1270 ELSE 960
960 CALL VCHAR(C,E,36)
970 CALL VCHAR(C,E,37)
980 CALL VCHAR(C,E,104)
990 CALL SOUND(340,425,2,-6,1)
1000 CALL KEY(0,K,S)
1010 IF S=0 THEN 1190
```



CHRISTINE COMPUTING

TI 99/4A

BLAST IT

Disarm bombs with a wire guided robot, bombs must be disarmed in order within a time limit. 5 levels of play, high score table.

£4.95

CORE!

A three part game, move left and right as you descend through the clouds, birds and aeroplanes. On to the caves and caverns then through the 3-D maze to the Core.

£4.95

FRUIT MACHINE

One arm bandit simulation featuring random hold, nudge and gamble. 'Reels' are fixed and can be learnt but motion is random.

£4.95

UFO

Blast the aliens as they appear from hyper-space, features 3 levels of play, refuel etc.

£4.95

SPUDZ

Score points as you collect potatoes without getting caught in the shadow cast by the setting sun or falling into your trench.

£4.95

COMMODORE 64

MUNCHER

100% machine code game making good use of the sound and graphic facilities of the 64. Munch your way around the maze avoiding the ghosts, etc.

£6.95

Send Cheques and Postal Orders to:

CHRISTINE COMPUTING, 6 FLORENCE CLOSE,
WATFORD, HERTS WD2 6AZ

Tel: Watford (0923) 672941

Please add 55p P&P per order.

Mind the flak, chaps. We must destroy the dam

Your aircraft is trying to destroy an enemy dam by dropping bouncing bombs. Your bombs must hit the water before hitting the dam or the enemy will retaliate with anti-aircraft fire, damaging your engines.

Five hits on the dam are required and you have up to 10 bombs to complete your mission. The game is fairly user-hostile in that there are many ways of losing — crashing into the water, getting hit by your own bomb, being shot down or just running out of bombs.

Full instructions are included in the program, so good luck and come back safely, chaps.

How it works

190-390 define characters and assign colours
400-590 title screen and instructions request
600-700 instructions
710-820 skill levels
830-1080 set up screen and set variables
1100-1170 set plane moving
1190-1210 detect plane crash
1230-1490 detect keys and release bomb
1500-1530 check position of bomb
1550-1790 anti-aircraft fire, hits on dam
1810-2140 dam destroyed
2170-2580 lose sequences
2600-2630 print routine

The pride of the squadron rests with you, your crew and your TI-99/4A. Type in Graham Baldwin's game for the standard TI... and return home safely to Blighty



Hints on conversion

TI BASIC is fairly standard but has some unique graphics commands, some of which are detailed below. The screen is made up of 32 columns and 24 rows.

CALL CLEAR clears the screen. Replace with CLS or similar on other machines

CALL CHAR re-defines an ASCE character with a sixteen digit hex string derived from an 8x8 grid

CALL HCHAR(R,C,ASCE) places a character at row R and column C on the screen and optionally repeats it horizontally

CALL VCHAR similar to HCHAR but repeats character vertically

CALL GCHAR(R, C, X) repeats the ASCE code to variable X of screen position R and C. Similar to a screen PEEK on other machines

CALL KEY(3, K, S) places the code of the key pressed into the variable K. Similar to GET on other computers

Main variables

ROW row of plane
COL column of plane
BR row of bomb
BC column of bomb
SCR number of hits on dam
BMS number of bombs
REV climb or dive plane

```
100 REM
110 REM ***DAM RAIDER***
120 REM
130 REM BY
140 REM GRAHAM BALDWIN
150 REM
160 CALL CLEAR
170 REM *DEF CHARS*
180 CALL SCREEN(5)
190 CALL CHAR(136,"101010FF10101010")
200 CALL CHAR(112,"FFFFFFFFFFFFFFFF")
210 CALL CHAR(113,"10387CFEFFFFFFFF")
220 CALL CHAR(97,"20A0C8FF7E081020")
230 CALL CHAR(98,"000020702")
240 CALL CHAR(105,"995A3CFF183C5A99")
250 CALL CHAR(120,"FFFFFFFFFFFFFFFF")
260 CALL CHAR(121,"FFFFFFFFFFFF3C3C3C")
270 CALL CHAR(122,"000103060C183060")
280 CALL CHAR(152,"3C7EFFFFFFFF7E3C")
290 REM *COLOURS*
300 CALL COLOR(14,12,2)
310 CALL COLOR(16,12,5)
320 CALL COLOR(11,8,5)
330 CALL COLOR(9,16,5)
340 CALL COLOR(10,9,16)
350 CALL COLOR(12,2,5)
360 REM *SET SCREEN*
370 FOR I=2 TO 8
```

```
380 CALL COLOR(1,16,5)
390 NEXT I
400 RANDOMIZE
410 CALL SOUND(100,1800,5)
420 MS$="DAM RAIDER"
430 RP=10
440 F=10
450 GOSUB 2590
460 MS$="*****"
470 CALL SOUND(100,1800,5)
480 RP=12
490 F=10
500 GOSUB 2590
510 MS$="DO YOU NEED BRIEFING? (Y/N)"
520 CALL SOUND(100,1800,5)
530 RP=16
540 F=3
550 GOSUB 2590
560 CALL KEY(3,K,S)
570 IF S=0 THEN 560
580 IF K=78 THEN 710
590 IF K<>89 THEN 560
600 CALL CLEAR
610 PRINT "YOU ARE FLYING A TWIN-ENGINE":
```


TI-99/4A PROGRAM

```

620 PRINT "DAM-RAIDER AIRCRAFT." :
630 PRINT "DESTROY THE DAM BY BOUNCING " :
640 PRINT "BOMBS OFF THE WATER." :
650 PRINT "FIVE HITS ARE REQUIRED." :
660 PRINT "A DIRECT HIT WILL CAUSE" :
670 PRINT "ANTI-AIRCRAFT FIRE TO" :
680 PRINT "DAMAGE ONE OF YOUR ENGINES." :
690 PRINT "PRESS U TO CLIMB AND" :
700 PRINT "SPACE TO DROP A BOMB." :
710 PRINT "SKILL LEVEL? (1-3)"
720 CALL SOUND(100,1800,6)
730 CALL KEY(3,K,S)
740 IF S=0 THEN 730
750 IF K<>49 THEN 780
760 BMS=10
770 GOTO 830
780 IF K<>50 THEN 810
790 BMS=7
800 GOTO 830
810 IF K<>51 THEN 720
820 BMS=5
830 ENG=2
840 CALL CLEAR
850 SCR=0
860 REM *SET UP SCREEN*
870 CALL HCHAR(20,1,113,29)
880 CALL HCHAR(21,1,112,128)
890 F=2
900 RP=23
910 CALL VCHAR(16,30,136,9)
920 CALL VCHAR(17,31,136,8)
930 CALL VCHAR(16,32,136,9)
940 CALL HCHAR(19,1,120,6)
950 CALL HCHAR(18,4,121)
960 CALL HCHAR(17,4,122)
970 CALL HCHAR(1,7,152)
980 CALL SOUND(10,1800,6)
990 MS$=" BOMBS- "&STR$(BMS)
1000 GOSUB 2600
1010 ROW=INT(10*RND)+1
1020 REV=1
1030 CALL HCHAR(20,1,113,29)
1040 COL=1
1050 F=10
1060 RP=23
1070 MS$=STR$(BMS)&" "
1080 GOSUB 2600
1090 REM *THE GAME*
1100 CALL HCHAR(ROW,COL,97)
1110 IF BMS=0 THEN 2290
1120 CALL KEY(3,K,S)
1130 CALL HCHAR(ROW,COL,32)
1140 COL=COL+1
1150 ROW=ROW+REV
1160 IF ROW>1 THEN 1190
1170 ROW=1
1180 REM *DETECT HITS*
1190 CALL GCHAR(ROW,COL,CRH)
1200 IF CRH<>32 THEN 2200
1210 IF COL=32 THEN 1010
1220 REM *DETECT KEYS*
1230 IF K<>85 THEN 1250
1240 REV=-1
1250 IF K<>32 THEN 1100
1260 CALL SOUND(15,1000,1,-7,8)
1270 BMS=BMS-1
1280 A=1
1290 Q=0
1300 BR=ROW+A+1
1310 BC=1
1320 BC=COL
1330 CALL HCHAR(BR,BC,98)
1340 CALL HCHAR(ROW,COL,32)
1350 COL=COL+1
1360 IF COL=32 THEN 1370 ELSE 1450
1370 COL=1

```

```

1380 REV=1
1390 CALL HCHAR(20,1,113,29)
1400 ROW=INT(10*RND)+1
1410 F=10
1420 RP=23
1430 MS$=STR$(BMS)&" "
1440 GOSUB 2600
1450 CALL HCHAR(ROW,COL,97)
1460 CALL HCHAR(BR,BC,32)
1470 BR=BR+A
1480 BC=BC+1
1490 IF BC>32 THEN 1100
1500 CALL GCHAR(BR,BC,H)
1510 IF H=32 THEN 1330 ELSE 1520
1520 IF H=113 THEN 1530 ELSE 1590
1530 A=-1
1540 REM *SHELL-FIRE*
1550 Q=1
1560 CALL SOUND(40,-6,1)
1570 CALL HCHAR(BR,BC,32)
1580 GOTO 1330
1590 IF H=136 THEN 1610 ELSE 1600
1600 IF H=97 THEN 2370
1610 IF Q=0 THEN 1620 ELSE 1750
1620 CALL HCHAR(16,5,105)
1630 CALL HCHAR(11,18,105)
1640 CALL HCHAR(11,27,105)
1650 CALL HCHAR(11,9,105)
1660 CALL SCREEN(9)
1670 CALL SOUND(500,110,8,-7,1)
1680 CALL SCREEN(5)
1690 CALL HCHAR(11,9,32)
1700 CALL HCHAR(11,18,32)
1710 CALL HCHAR(11,27,32)
1720 CALL HCHAR(16,5,32)
1730 ENG=ENG-1
1740 IF ENG=0 THEN 2460 ELSE 1100
1750 SCR=SCR+1
1760 CALL HCHAR(BR,BC,105)
1770 CALL SOUND(500,110,5,220,5,-7,1)
1780 CALL HCHAR(BR,BC,32)
1790 IF SCR=5 THEN 1800 ELSE 1100
1800 REM *WIN SEQ*
1810 CALL VCHAR(16,31,105,9)
1820 CALL SCREEN(11)
1830 CALL HCHAR(ROW,COL,32)
1840 CALL SOUND(100,110,4,-7,1)
1850 CALL SCREEN(5)
1860 FOR I=1 TO 16
1870 CALL SOUND(-150,-7,1)
1880 NEXT I
1890 CALL VCHAR(16,30,32,9)
1900 CALL VCHAR(16,31,32,9)
1910 CALL VCHAR(16,32,32,9)
1920 CALL HCHAR(21,1,112,128)
1930 CALL HCHAR(19,1,32,6)
1940 CALL HCHAR(18,4,32)
1950 CALL HCHAR(17,4,32)
1960 FOR I=20 TO 24
1970 CALL HCHAR(I,1,113,32)
1980 FOR D=1 TO 200
1990 NEXT D
2000 CALL HCHAR(I,1,32,32)
2010 CALL SOUND(300,-5,6)
2020 NEXT I
2030 CALL CLEAR
2040 CALL SOUND(100,294,4,440,4,370,4)
2050 CALL SOUND(1000,294,1,440,1,370,1)
2060 PRINT "YOU BURST THE DAM." :
2070 PRINT "MISSION SUCCESSFUL." :
2080 PRINT "ANOTHER GAME? (Y/N)"
2090 CALL KEY(3,K,S)
2100 IF S=0 THEN 2090
2110 IF K=89 THEN 2130
2120 IF K=78 THEN 2450 ELSE 2090
2130 CALL CLEAR

```


TI-99/4A PROGRAM

```

2140 GOTO 710
2150 CALL CLEAR
2160 REM *LOSE SEQ*
2170 CALL SCREEN(5)
2180 PRINT " YOU WERE SHOT DOWN."
2190 GOTO 2080
2200 CALL HCHAR(ROW,COL,105)
2210 CALL SCREEN(9)
2220 CALL SOUND(990,110,7,-7,1)
2230 CALL CLEAR
2240 CALL SCREEN(5)
2250 PRINT " YOU CRASHED."
2260 CALL SOUND(100,196,1)
2270 CALL SOUND(400,165,1)
2280 GOTO 2080
2290 CALL CLEAR
2300 CALL SOUND(100,196,1)
2310 CALL SOUND(400,165,1)
2320 PRINT " YOU HAVE DROPPED"
2330 PRINT " ALL YOUR BOMBS."
2340 PRINT " MISSION ABANDONED."
2350 GOTO 2080
2360 CALL CLEAR
2370 CALL HCHAR(ROW,COL,105)
2380 CALL SCREEN(9)
2390 CALL SOUND(600,110,5,220,5,-7,1)
2400 CALL SCREEN(5)
2410 CALL CLEAR
2420 PRINT " YOU WERE BROUGHT DOWN"
2430 PRINT " BY YOUR OWN BOMB."
2440 GOTO 2080
2450 END
2460 CALL HCHAR(ROW,COL,32)

```

```

2470 ROW=1
2480 COL=1
2490 CALL HCHAR(ROW,COL,97)
2500 CALL SOUND(-50,-7,1)
2510 CALL HCHAR(ROW,COL,32)
2520 ROW=ROW+1
2530 COL=COL+1
2540 CALL GCHAR(ROW,COL,CR)
2550 IF CR=32 THEN 2490
2560 CALL SCREEN(9)
2570 CALL SOUND(500,110,2,-7,0)
2580 GOTO 2150
2590 REM ** SCORE ROUTINE**
2600 FOR I=1 TO LEN(MS$)
2610 CALL HCHAR(RP,F+I,ASC(SEG$(MS$,I,1)))
2620 NEXT I
2630 RETURN

```



48K
SPECTRUM

Regardez!

BBC
acorn
electron

LANGUAGE LEARNING AIDS FOR FRENCH, GERMAN & SPANISH

Already in use in numerous schools and colleges, these programs provide an immensely powerful aid to foreign language learning. The cassettes include extensive vocabulary lists arranged as a series of lessons, each covering a different subject. A tuition control program enables individual lessons to be loaded and used as required.

Words, phrases etc are presented first in one language, then the other complete with all necessary accents and special

characters. Masculine, feminine and neuter words appear in different colours to encourage gender learning. All lessons can be run in three different ways, i.e. learning only, self-test or speed and accuracy test.

The programs are suitable for pupils of all ages as simple commands enable new lessons in vocabulary or grammar to be entered by the user. These may then be edited as required and stored on cassette for later use. Invaluable for homework and exam revision!

Level A Cassettes: Contain the tuition control program and 16 comprehensive lessons for general vocabulary learning.

Level B Cassettes: Contain the tuition control program and 16 lessons including verb lists, adjectives, adverbs and phrases.

Available from dealers, large stores or mail order.

Also Available "THE SPANISH TUTOR"

KOSMOS Software

1 Pilgrims Close, Harlington, Dunstable, Beds. LU5 6LX Tel: 05255 3942

Kosmos Software, 1 Pilgrims Close, Harlington, Dunstable, Beds LU5 6LX.

Please supply the following programs for the computer (BBC/SPECTRUM/ACORN ELECTRON) (Prices include postage & packing)

The French Mistress Level A @ £9.95	The French Mistress Level B @ £9.95
The German Master Level A @ £9.95	The German Master Level B @ £9.95
The Spanish Tutor Level A @ £9.95	The Spanish Tutor Level B @ £9.95

Mr/Mrs/Miss.....
Address.....

Post code..... HCW52
I enclose a cheque postal order value £..... payable to Kosmos Software



LETTERS

Books: the good... and the disasters

I have taken your magazine for the last five months and I am very interested in comments on programs that do not work from listings in books and magazines. I own a VIC-20 and several of my friends and relations have an assortment of computers so that between us we can solve most problems.

I bought a book by Tim Hartnell, published by Interface Publications, called *Symphony for a Melancholy Computer: 50 Programs for a VIC-20*. It stated that all the listings had been checked. They all worked perfectly. Any that did not at first was the result of errors on my part. When corrected they all worked.

A short time later I bought another book, *Sixty Programs for the VIC-20*, published by Pan Books. This book is a disaster.

I now have on tape 29 programs from the book for the VIC up to 16K. Not one works without your having to spend hours and gaining help from friends to make even the simple ones appear on the screen.

Six times the computer has gone away and sulked and twice the phrase "FORMULA TOO COMPLEX" has appeared.

To make sure that I had not made any mistakes, other people have taken the book and tried the listings on their VIC-20s. Still no good.

My son has a Commodore 64 and he saw my book before I had entered any of the listings and went out and bought a similar book, also from Pan, for the 64. Another disaster.

What comments do your readers care to make? Any recommended books that work will be appreciated.
P.R. Taylor, Northfield, Birmingham

A spokeswoman for Pan Books said: "There is a massive error in both books which makes many of the programs impossible to use."

"This has been extensively advertised and bookshops have been told."

"We are asking people who bought the books to send a proof of purchase — like the title page — and they will receive a reprinted copy of their book in a few weeks."

"These books were a joint venture between ourselves and another company which provided the programs from several authors."

Pan Books is at 18-20

Send your letter to Letters, Home Computing Weekly, No.1 Golden Square, London W1R 3AB. Don't forget to name your computer — the best letter could win £5-worth of software. Got a problem with your micro? We'll soon be starting a queries page, so send your technical questions to Queries at the above address



Cavaye Place, London EC4A 1AB

• Mr Taylor also mentions magazine listings and readers may like to know how these are prepared for publication in Home Computing Weekly.

Programs from readers are accepted on cassette only — we do not publish contributors' own listings — and then tested for quality in our computer room.

Those chosen for publication are printed out by us and reproduced photographically in the magazine.

Microdrive, Sir Clive

I read with interest Stephen Butcher's letter in HCW 49. I have had my Spectrum 48K since October 1982 with a promise that I would get an order form for the ZX Microdrives as soon as they are available. After letters not answered and phone calls — the last one a week past — I'm still waiting. Here's a letter for Sir Uncle Clive:

Dear Uncle Clive

You sent me last year a lovely leaflet, telling me I would soon receive an order form for a ZX Microdrive. Alas! It has never been received!

Alas! I seek it here
I seek it there
I seek it in every way
I try by letter
I try by phone
and a little bird says "Sinclair Research"
Please hold the line
We will be with you in ten minutes time
or was it "Apple-Blossom time"!

We seek it here. We seek it there. We seek that elusive microdrive.

Be it on the launch pad or up in space. That elusive interface.

I look around and it is not in this place

Dear Uncle Clive, when will it be available?
A.J. Russell, Street, Somerset

More Texas phone lines

Here are some additional lines for the T I-99/4A Texas Telephones program (HCW 49) which readers may find amusing.

Well done, HCW, for your excellent magazine and for supporting the TI — they both give my family hours of fun. The TI is an excellent computer. Ours will never be discarded and left to rot in the attic. It will be worn out with constant use long before then.

Keep up the good work.

50 GOSUB 5080
675 GOSUB 5180
5080 CALL CLEAR
5090 PRINT "WHAT IS YOUR NAME?"
5095 PRINT " "; NS
5110 IF NS = " " THEN 5090

5120 PRINT " " ; : : : :
5130 PRINT "WHAT IS YOUR PHONE NUMBER?"

5140 PRINT " "; TS
5150 INPUT TS
5160 IF TS = " " THEN 5130

5170 RETURN
5180 PRINT "HELLO, IS THAT "; TS

5185 PRINT " " ; : :
5190 PRINT "COULD I SPEAK TO "; NS

5200 FOR DELAY = 1 TO 2000

5210 NEXT DELAY
5220 CALL CLEAR
5230 RETURN

Maureen Ashworth, Accrington, Lancs

Letter letter

As printed, the Spectrum Renumber program in HCW 40 does not work. However, only a minor change is necessary to make it an excellent program.

Change the first line to "9500 LET Z=0: REM Renumber by N DORE" and all will be well.

For the technically minded, the program needs to have Z as the first variable in the area of memory reserved for variables. Variables are stored in order of first occurrence. As the program is listed the GOTO 9980 in line 9501 means that the n in the FOR-NEXT loop is the first variable used. When line 9620 looks for the value of Z in the variable area it finds the last value of n and so corrupts GOTO and GOSUB line numbers.

N.W. Jinks, Sutton, Surrey

You're in the Gunners and the enemy is firing back...

Ideally suited to TI BASIC, my Artillery game can use its high-resolution graphics and multiple graphics without being made useless by its speed limitations.

There are two heavy artillery guns on a random piece of terrain — two level plains with a plateau between them.

Each gun fires in turn and has to hit the opposing gun to win.

Shell trajectory is determined by launch angle and speed, gravity and wind velocity, again random, but constant throughout the game.

Shells cannot pass through the central plateau or any of the ground squares.

There is a short delay at the beginning while display lists, and so on, are set up. Then the terrain, guns and titling are printed.

When the green prompt square appears at top left, enter muzzle velocity and angle of elevation, pressing enter after each.

Mistakes can be corrected provided the enter key has not been pressed. To do this, press D — the cursor right key — and the right-most digit will be erased.

When the guns are fired the

You'll need careful calculations to win at P.D. Myring's two-player Artillery game. It runs on the unexpanded TI-99/4A

shell path appears on the screen and explosions occur when either the ground or a gun have been hit.

Old shell paths remain on the

screen as a guide, but some will be over-written by subsequent firing.

The blue gun on the left fires first and, as the first player to

```

100 REM ARTILLERY
110 REM *****

120 CALL CLEAR
130 PRINT " ONE MOMENT PLEASE ..... "
140 GOSUB 1900
150 CALL CLEAR
160 GOSUB 2400
170 REM PLAY
180 PLYR=PLYR
190 XENT=14.5-10.5*PLYR
200 MAX=4
210 GOSUB 1170
220 MZ=VL
230 XENT=19.5-10.5*PLYR
240 MARK=3
250 GOSUB 1170
260 ANGLE=VL
270 YENT=YENT+(1-PLYR)/2
280 XVMZ=ZCOS(ANGLE*(4*PI/180)/100)
290 YVMZ=ZSIN(ANGLE*(4*PI/180)/100)
300 IF PLYR=1 THEN 360
310 DIR=136
320 XVM=XV
330 XG=XG2
340 YG=YG2
350 GOTO 390
360 XG=XG1
370 YG=YG1
380 DIR=112
390 XP=XG
400 YP=YG
410 CALL SOUND(50,-5,2)
420 CALL SOUND(-100,-6,10)
430 FOR T=1 TO 50 STEP 2
440 XOLD=XP
450 YOLD=YP
460 XP=XG+(.002*(XOLD-XP)*(1-EXP(-.01*T)))/.0001+30*XT)
470 YP=YG+(.002*(YOLD-YP)*(1-EXP(-.01*T)))/.0001-32*YT/.01)
480 IF INT(YP)<1 THEN 660
490 IF YOLD>YP THEN 510
500 CALL SOUND(-1000,2000-10*YP,30-YP,1995-10*YP,30-YP,1995-10*YP,30-YP)
510 CHDIR=(INT(4*(XP-INT(XP)))+4*INT(4*(YP-INT(YP))))
520 REM HIT GROUND?
530 IF INT(XP)>32 THEN 670
540 IF INT(XP)<1 THEN 670
550 IF INT(XP)>PL THEN 580
560 IF INT(YP)>YG1 THEN 1070
570 GOTO 620
580 IF INT(XP)>PR THEN 610
590 IF INT(YP)>PT THEN 710
600 GOTO 620
610 IF INT(YP)>YG2 THEN 970
620 CALL GCHAR(INT(YP),INT(XP),G)
630 IF G>111 THEN 650
640 IF G>32 THEN 660
650 CALL HCHAR(INT(YP),INT(XP),CH)
660 NEXT T
670 REM NEXT
680 GOTO 100
690 END

700 REM EXPLOSION
710 XEX=XP+(PT-YP)*(XOLD-XP)/(YOLD-YP)
720 IF INT(XEX)>PR THEN 740
730 IF INT(XEX)<PL THEN 750
740 IF XOLD<XP THEN 890 ELSE 810
750 CALL HCHAR(PT-1,INT(XEX),50)
760 CALL HCHAR(PT,INT(XEX),104)
770 GOSUB 1020
780 CALL HCHAR(PT-1,INT(XEX),32)
790 CALL HCHAR(PT,INT(XEX),105)
800 GOTO 670
810 YEX=YP+(PR-XP)*(YOLD-YP)/(XOLD-XP)
820 IF INT(YEX)>YG2 THEN 970
830 CALL HCHAR(INT(YEX),PR+1,63)
840 CALL HCHAR(INT(YEX),PR,104)
850 GOSUB 1020
860 CALL HCHAR(INT(YEX),PR+1,32)
870 CALL HCHAR(INT(YEX),PR,107)
880 GOTO 670
890 YEX=YP+(PL-XP)*(YOLD-YP)/(XOLD-XP)
900 IF INT(YEX)>YG1 THEN 1070
910 CALL HCHAR(INT(YEX),PL-1,59)
920 CALL HCHAR(INT(YEX),PL,104)
930 GOSUB 1020
940 CALL HCHAR(INT(YEX),PL-1,32)
950 CALL HCHAR(INT(YEX),PL,106)
960 GOTO 670
970 XEX=XP+(YG2+1-YP)*(XOLD-XP)/(YOLD-YP)
980 YEX=YG2
990 CALL GCHAR(YG2,INT(XEX),Q)
1000 IF Q=40 THEN 1540
1010 CALL HCHAR(YG2,INT(XEX),50)
1020 CALL HCHAR(YG2+1,INT(XEX),104)

```

Hints on conversion

TI BASIC is fairly standard with two main exceptions. Single statement lines are used and subroutines for graphics and sound are available, prefixed with CALL, as detailed below.

CALL CLEAR clears screen

CALL SCREEN defines screen colour

CALL CHAR defines a character with a 16-digit hex string from an eight by eight grid. Other machines will use a different method

CALL COLOUR (character set, foreground, background) sets colour of character

CALL HCHAR (row, column, character code, number of repeats) places character on screen at co-ordinates specified and optionally repeats character horizontally

CALL VCHAR similar to HCHAR but repeats character vertically

CALL GCHAR (row, column, variable) similar to a screen PEEK on other machines. Gives the ASCII code at the given screen co-ordinates

CALL SOUND (D,F,V) gives sound of duration D, frequency F and volume V. Three sounds can be produced at once

How it works

140 GOSUB to set up colours, characters

160 GOSUB to set up screen — terrain, cannons, etc

170 game start

190-260 player enters speed and angle data

210-250 GOSUB to "entry" subroutine

280-290 convert speed and angle data into X & Y velocities

300-400 select correct gun and firing position

430-660 main loop — determine behaviour of shell in flight

460-470 the crucial lines — formula for path of flight

510 determine which shell-path character is printed

520-610 determine if shell has hit ground, and if so, where. Incorporates 700-1160

620-660 print next shell-path character

700-1160 detects where shell has hit ground, and which explosion and crater characters to use.

710,810,890,970 & 1070 all formulae to find exact point at which shell hits ground/hill. Then the correct nearest square can be used for explosion and crater.

1170-1470 subroutine for entering shell speed and angle data

1250 detects enter key pressed

1260 detects D key pressed

1270 & 1280 rejects keys pressed which are not numerals

1290-1360 shift all existing numbers to left, and prints new one

1370-1430 shift all existing numbers to right and eliminates LSD — for entry editing

1480-1520 print string label at points specified by XLB and YLB

1530-1800 direct hit routine. Prints explosion, plays fanfare etc

1810-1870 subroutine for normal explosion sound

1880-2080 subroutine sets up colours

2090-2380 subroutine sets up characters

2390-2870 determines random shape of terrain. Prints terrain, guns, etc

2510 determines random wind speed and direction

score a direct hit wins, it has the advantage.

So players could take alternate ends. Perhaps the last loser or least experienced player could get blue.

Useful muzzle velocity is 500-1,100 feet per second and if strong winds are blowing with you it's best to fire high.

TI-99/4A PROGRAM

```

1030 GOSUB 1020
1040 CALL HCHAR(YG2,INT(XEX),32)
1050 CALL HCHAR(YG2+1,INT(XEX),105)
1060 GOTO 670
1070 XEX=XE+(YG1+1-YE)*(XOLD-XE)/(YOLD-YE)
1080 YE=YG1
1090 CALL GCHAR(YG1,INT(XEX),Q)
1100 IF Q=36 THEN 1540
1110 CALL HCHAR(YG1,INT(XEX),50)
1120 CALL HCHAR(YG1+1,INT(XEX),104)
1130 GOSUB 1020
1140 CALL HCHAR(YG1,INT(XEX),32)
1150 CALL HCHAR(YG1+1,INT(XEX),105)
1160 GOTO 670
1170 REM ENTRY
1180 CALL SOUND(200,400,20)
1190 CALL SOUND(400,300,20)
1200 VL=0
1210 CALL HCHAR(YENT,XENT+1,104)
1220 CALL KEY(0,KY,S)
1230 IF S=0 THEN 1220
1240 CALL SOUND(10,1000,10)
1250 IF KY=13 THEN 1440
1260 IF KY=68 THEN 1370
1270 IF KY=48 THEN 1220
1280 IF KY=57 THEN 1220
1290 FOR N=XENT-MAX+1 TO XENT-1
1300 CALL GCHAR(YENT,N+1,Q)

```

```

1310 CALL HCHAR(YENT,N,Q)
1320 NEXT N
1330 VL=VL+10
1340 CALL HCHAR(YENT,XENT,KY)
1350 VL=VL+KY-40
1360 GOTO 1220
1370 FOR N=XENT-1 TO XENT-MAX+1 STEP -1
1380 CALL GCHAR(YENT,N,Q)
1390 CALL HCHAR(YENT,N+1,Q)
1400 NEXT N
1410 CALL HCHAR(YENT,N+1,32)
1420 VL=INT(VL/10)
1430 GOTO 1220
1440 CALL HCHAR(YENT,XENT+1,32)
1450 FOR N=1 TO 100
1460 NEXT N
1470 RETURN
1480 REM LABEL
1490 FOR N=1 TO LEN(LB)
1500 CALL HCHAR(YLB,XLB+N-1,ASC(SEG$(LB$,N)))
1510 NEXT N
1520 RETURN
1530 REM BIG BANG
1540 CALL VCHAR(INT(YE)-1,INT(XE),56,2)
1550 FOR EXPL=0 TO 30 STEP 2
1560 CALL SCREEN(EXPL/2+1)
1570 CALL SOUND(-1000,200,EXPL,-7,EXPL)
1580 FOR DELAY=1 TO 20
1590 NEXT DELAY
1600 NEXT EXPL
1610 CALL SCREEN(SCRN)
1620 REM PANPANE
1630 DATA 200,0,16,20,2,1175,1,1047,1,1047,6,1175
1640 DATA 1,1047,704,1045,1,1175,800,1173,1,1319,900,1317,1,1175,800,1173,1,1047
1650 RESTORE 1630
1660 READ X,V1,V2,V3
1670 FOR NT=1 TO 4
1680 READ DUR,TOPE
1690 CALL SOUND(DUR,X,TOPE,V1)
1700 NEXT NT
1710 FOR NT=1 TO 6
1720 READ DUR,TN1,TN2,TN3
1730 CALL SOUND(DUR,X,TN1,V1,TN2,V2,TN3,V3)
1740 NEXT NT
1750 XLB=2
1760 YLB=20
1770 LB$="TO PLAY AGAIN, PRESS ANY KEY."
1780 GOSUB 1400
1790 CALL KEY(0,KY,ST)
1800 IF ST=0 THEN 1790 ELSE 150
1810 REM BANG!
1820 FOR EXPL=0 TO 30 STEP 5
1830 CALL SOUND(-200,-7,EXPL,110,EXPL)
1840 FOR DELAY=1 TO 15
1850 NEXT DELAY
1860 NEXT EXPL
1870 RETURN
1880 REM SET UP COLORS & SPECIAL CHARACTERS.
1890 REM COLORS
1900 BACK=2
1910 SCR=2
1920 CALL COLOR(1,BACK,BACK)
1930 CALL COLOR(2,10,BACK)
1940 CALL COLOR(3,14,BACK)
1950 CALL COLOR(4,14,BACK)
1960 CALL COLOR(5,8,BACK)
1970 CALL COLOR(6,8,BACK)
1980 CALL COLOR(7,8,BACK)
1990 CALL COLOR(8,8,BACK)
2000 CALL COLOR(9,6,BACK)
2010 CALL COLOR(10,13,BACK)
2020 CALL COLOR(11,5,BACK)
2030 CALL COLOR(12,5,BACK)
2040 CALL COLOR(13,5,BACK)
2050 CALL COLOR(14,7,BACK)
2060 CALL COLOR(15,7,BACK)
2070 CALL COLOR(16,7,BACK)
2080 CALL SCREEN(SCRN)
2090 REM GUNS
2100 CALL CHAR(40,"47EE7C3076EFCF86")
2110 CALL CHAR(96,"E2773E1C6EF7F361")
2120 REM EXPLOSIONS
2130 CALL CHAR(50,"502980E7C807E10")
2140 CALL CHAR(59,"120A241F261C0010")
2150 CALL CHAR(63,"485024F864381000")
2160 REM GROUND & CRATERS
2170 CALL CHAR(104,"FFFFFFFFFFFFFF")
2180 CALL CHAR(105,"00000101C3C3FFFF")
2190 CALL CHAR(106,"7F3F1F0F0F1F7F")
2200 CALL CHAR(107,"FEFCF8F8F8F8FE")
2210 REM SHELL PATH
2220 FB$=""
2230 AB$="C0C0"
2240 BB$="3030"
2250 CB$="0C0C"
2260 DB$="8080"
2270 FOR N=1 TO 4

```

```

2280 CALL CHAR(108+4*N,FB$AB)
2290 CALL CHAR(132+4*N,FB$AB)
2300 CALL CHAR(109+4*N,FB$BB)
2310 CALL CHAR(133+4*N,FB$BB)
2320 CALL CHAR(110+4*N,FB$CB)
2330 CALL CHAR(134+4*N,FB$CB)
2340 CALL CHAR(111+4*N,FB$DB)
2350 CALL CHAR(135+4*N,FB$DB)
2360 FB$=FB$"0000"
2370 NEXT N
2380 RETURN
2390 REM TERRAIN
2400 RANDOMIZE
2410 XG1=3
2420 XG2=30
2430 TRNA=INT(20*RND)-10
2440 IF TRNA=0 THEN 2430
2450 TRNB=INT(RND*(17-ABS(TRNA)))+1
2460 TRNC=2*INT(4*RND)+2
2470 PL=16-TRNC/2
2480 PR=15-TRNC/2
2490 PT=24-ABS(TRNA)-TRNB
2500 CALL HCHAR(24,1,104,32)
2510 W=INT(200*RND)-100
2520 IF W<0 THEN 2550
2530 LB$="WINDH"&STR$(ABS(W))&"MPH"
2540 GOTO 2560
2550 LB$="WINDH"&STR$(ABS(W))&"MPH"
2560 YLB=24
2570 IF TRNA<0 THEN 2660
2580 FOR N=1 TO TRNA
2590 CALL HCHAR(24-N,1,104,PR)
2600 NEXT N
2610 XLB=2
2620 GOSUB 1490
2630 YG1=23-TRNA
2640 YG2=23
2650 GOTO 2730
2660 FOR N=1 TO (-TRNA)
2670 CALL HCHAR(24-N,PL,104,33-PL)
2680 NEXT N
2690 XLB=PL+1
2700 GOSUB 1490
2710 YG1=23
2720 YG2=23+TRNA
2730 FOR N=1 TO TRNB
2740 CALL HCHAR(24-ABS(TRNA)-N,PL,104,TRNC)
2750 NEXT N
2760 CALL VCHAR(YG1,XG1,96)
2770 CALL VCHAR(YG2,XG2,40)
2780 RANDOMIZE
2790 XLB=1
2800 YLB=1
2810 LB$="SPEED ANGLE"
2820 GOSUB 1490
2830 XLB=22
2840 GOSUB 1400
2850 PLYR=-1
2860 YENT=YLB+1
2870 RETURN

```

CALLING ALL AQUARIUS USERS!

Now there's a User Group especially for you. For just £6 or £12 you can join the AQUARIUS USER club and get a monthly magazine which gives you all the latest on your Aquarius, news, facts, and features.

**SPECIAL
OFFER**
Join before
May 31st and
**SAVE
£2**

AQUARIUS USER

AQUARIUS USER is packed with information written by experts. It covers details on new products, and peripherals, reviews on the latest software, general news about home computers, user tips, reader offers, competitions, letters and much more.

If you're part of the dawning age of AQUARIUS join AQUARIUS USER now: AQUARIUS USER LTD, 66 Wymering Road, London W9.

Name _____

Address _____

Postcode _____

Please enrol me as a member of AQUARIUS USER.

I enclose £6 subscription for 5 issues ☐ or £10 for one year ☐
payable to Aquarius User Ltd, 66 Wymering Rd, London W9

Access Card holders ring 01-289 2188 (24 hours)

TI-994A SOFTWARE REVIEWS

House of Horrors and Star Strike £5.99

Solid Software, 35 Melville Rd, Bispham, Blackpool FY2 9JF

House of Horrors is an adventure game with graphics. To start you must find the key to enter the house. Clues to the key are given on screen and once deciphered you are given an option of five levels of play and a chance to have the instructions displayed if required. Without warning you are beset by problems which must be solved before you can proceed. There is an egotistical gorilla who likes to hear his name, and this is printed on screen for a brief moment after which you must type in the name from memory. The game

proved to be difficult enough at level one so if you get to level five you're a better man than I Gunga Din. This game is excellent fun and a must for all you adventure fanatics.

Star Strike is an arcade style game where you are the pilot of a spaceship defending against alien invaders. The graphics used are excellent and represent the view through a cockpit window at the aliens coming towards you in a good 3D effect. Although the action is limited, the game is well presented with excellent graphics. The whole package adds up to a very good value for money.

instructions	80%
playability	95%
graphics	95%
value for money	95%



Fun, games and utilities for the Texas

Play golf and cards, have adventures and hit the aliens — even design your own graphics with these Texas programs

Texi Golf £3.50

SofTI, 14 Station Rd, Brough, North Humberside

The game of golf is now available for most micros and the TI is no exception, with several versions on the market.

Unfortunately, most of the realism is taken out of this program, with non-standard clubs used when it would have been just as easy to use a standard set.

Instead of the usual three woods, nine irons and a putter, only 10 clubs are used. These are as follows: seven irons, one driver, one rather old-fashioned brassie and one wedge.

The course has nine holes, and any hole can be played at any time. I was surprised to find that there were no lakes, streams or

trees which are almost essential features of a real course.

Rather a strange aspect of the game was that the computer chooses a "suitable" club if you land in the rough. This turns out to be extremely annoying, as nine times out of ten you have to play away from the green in order to avoid going into the rough again.

After the ninth hole, your score card and handicap are displayed.

This is not the best version of golf I have seen, but it could be greatly improved with just a few minor additions.

instructions	70%
playability	60%
graphics	60%
value for money	85%



Napoleon £4.95

MarKat, 4 Cloreen Pk, Belfast BT9 5BX

A card game, very similar to Whist, in which the player competes against the computer.

When the cards have been dealt (the player and the computer take turns), the non-dealer states how many tricks he thinks he can win. (As only five cards are dealt, this number can be no higher than five and no less than two). The dealer may then either pass or make a higher bid.

The first card is played by whoever makes the highest bid, and the suit of this card then becomes the trump suit.

Following suit, if possible, the opponent then plays a card. If the value of this card is higher than that played by the other

contestant, the opponent wins the trick. As in Whist, if a trump is played it wins the trick.

Unlike Whist, there is a scoring system which awards points to the players, according to the amount of tricks.

The cards are displayed extremely well on the screen, with the player's cards face up, and the computer's face down.

Although the computer knows which cards the player holds, the instructions say it does not use this information to its own advantage.

instructions	95%
playability	75%
graphics	90%
value for money	80%



Graphics Creator & Screen Editor £5.95

PiKaDee, 35 Parket St, Preston, Lancs PR2 2AH

Gone are the days when I used to spend many a boring evening scribbling on a grotty piece of over-used graph paper, attempting to design yet another alien for the world's billionth version of Space Invaders. All I have to do now is load Graphics Creator, and after tapping in my requirements all the hexadecimal coding is worked out for me. Very convenient.

As well as Graphics Creator and Screen Editor there are three other programs containing character sets, but I found these rather hard to load.

Upon execution of the program, an 8 x 8 character grid is displayed on the screen, together with the letters A to Z adn, underneath them, two rows of un-defined characters.

To design your own character, all you have to do is tap in the coodinates of the squares you need filling. Characters can be rotated, inverted, stored as one as the user-defined characters, and saved on cassette. Their hex listings can also be shown on the screen.

The screen editor allows you to design a screen using your pre-defined characters, which is very useful when planning a title screen for a program. A must for all TI owners.

instructions	100%
ease of use	95%
display	95%
value for money	95%



Fun pac 2 £6.95

Virgin, 61-63, Portobello Rd, London W11 3DD

A package consisting of three games, one arcade style and two text-only adventure games.

The arcade game has the rather topical title of Escape the Mugger, and you are the rather hapless victim who has to escape. The villain is, however, extremely fleet footed and you must exercise a great deal of cunning in order to avoid being another mugging statistic.

Starship Supernova is a text only adventure, where you are the captain of a space salvage ship which happens upon an alien spaceship heading for a supernova and destruction. The entire game is played against the clock and when time starts to run

out you are given visual warnings to the effect that you are nearing the supernova.

The final game on the tape is also text only, and places you in as a gunfighter in the old wild west.

Different factors are given to help you decide on which action to take and your skill will decide whether you clean up the town or end up dead.

Three games on this tape are very well presented and I found them entertaining and good fun to play. At the combined price of £6.95, good value for money.

instructions	90%
playability	90%
graphics	80%
value for money	90%



Hoppitt £4.95

Warwick, 40 Kingsway, Newby, Scarborough, Yorks

This otherwise good Frogger-type game is marred by the fact that the BREAK key has been disabled, so you have to make sure you write out the instructions (and watch out for fall-through due to some poor keyboard scans). To re-start you have to quit and reload. This can take quite a few minutes, so unless you know the way round the protection facility of Extended BASIC and can therefore re-instate BREAK you will have to put up with it.

A feature of the initialisation is that you can elect to choose your own keys. The display is

quite good, although some characters look a little anorexic.

There are a number of hazards while conducting 10 frogs in turn across the road and river — for one the turtles and ducks can become bright red and treacherously frog-eating. A few extra items make this version different, but in the end it boils down to "who's got the highest score?" There are five levels, but the fastest is breakneck and I see no point in playing at such a speed. Needs Extended BASIC. Speech, joysticks optional. P.B.

instructions	80%
playability	80%
graphics	80%
value for money	50%



Radar Defence £5

Pauline Programs, from Stainless, 10 Alstone Road, Stockport

Another in a long line of programs which urge you to defend yourself against marauding aliens. Given the deficiencies of TI BASIC, the

graphics and keyboard scanning are above average, although the instructions are sparse but adequate.

There are three screens of aliens attacking you one at a time — this owes more to the slowness of TI BASIC than to any alien chivalry — and you control your defending base (one of three) through the keyboard. There is a force field which can be used a

Five more for the Texas

Useful and entertaining... our reviewers tell you what this batch of Texas tapes offers

maximum of three times to provide protection, and you can fire at and destroy the aliens, although some do fire back. There are additional hazards in the shape of invisible aliens, but it is fairly easy to find and destroy them, especially since some cannot resist firing and giving themselves away.

The "radar" is actually more of a gimmick than any real help, often pointing in the wrong direction. The aim is to score as

many points as you can but, as usual, once you stop playing and switch off, you'll have to rely on a friend to vouch for your ability. P.B.

instructions	70%
playability	90%
graphics	75%
value for money	65%



Fun Pac 3 £6.95

Virgin, 61-63 Portobello Rd, London W11 3DD

A good value package comprising three arcade-type games. Golf, offering one to 18 holes, has all the ingredients, such as type of iron and hazards like trees, water and sand bunkers. You select iron, direction and strength and then drive by pressing the space bar.

Nuclear Disaster made me think of other similarly-named games, but I found it to be different. You see the plan of a building giving positions of a leaking reactor, decontamination chamber and a room containing door sealing blocks. Scattered about are scientists for you to

rescue by pushing them, singly, into the decontamination chamber. Then you have to seal the building with the blocks before earth is contaminated.

Sea Wolf puts you in command of a submarine. With 20 torpedoes you have to destroy the enemy ships. You are given the view through a periscope and a radar scanner — dive fast if you see the destroyer.

With full instructions on a leaflet and on screen, I found this collection entertain and good fun. D.B.

instructions	90%
playability	90%
graphics	90%
value for money	85%



Floor Planner £3.50

B. Jackson, 21 Roman Way, New Balderton, Newark, Notts NG24 3AU

This falls into a category which contains all those programs unlikely to have a general application and appeal. If you are into floor planning in a big way then obviously any program dealing with that subject will help. This one uses sprites to allow you to select either items from a built-in library — radiators, power points, windows, doors, beds double and single, tables, chairs and settees — or name, but not define the shape of, your own objects, and to locate them in a room.

Relative sizes are given in metric units — good to see someone is using the more

sensible if unfamiliar system — and objects can be orientated to suit the room. The documentation is quite good, even listing the main routines and their function, a practice more software houses ought to adopt.

However, perhaps due to my own lack of aptitude, I found the program difficult to understand and operate, and although items were colour-coded, the block approach did not help me at all. Although the programmer has obviously put a lot of thought into this, his attentions might have been more fruitfully applied elsewhere. Needs Extended BASIC. P.B.

instructions	100%
ease of use	50%
display	70%
value for money	90%



Bar Graph £3.50

B. Jackson, 21 Roman Way, New Balderton, Newark, Notts NG24 3AU

Displaying data with histograms is usually easier when the bars are PRINTed on the screen. It can be quite a challenge to write a program to provide vertical bars with pixel precision. You

need a range of 'caps' for the bars and calculations must be performed.

This produces main bodies for the bars, then goes over them again adding the caps, which can look a little disconcerting. You can create up to 10 different 12-element histograms, each named by a letter, and each allowing you to title X and Y axes. You can enter values 0.02 — 9999999 and it will automatically rescale, choosing the

best range for presentation. A multiplication factor is displayed with each graph to indicate the relationship of the scale displayed to the actual axis. There are a number of options.

The calculations are done surprisingly quickly. It is possible to store your data on tape, and to retrieve it, creating a limited but powerful graphics presentation facility. If only the tape transfer rate was higher this would make a valuable graphics

tool for conferences etc. Needs Extended BASIC. P.B.

instructions	65%
ease of use	90%
display	90%
value for money	80%



NEW!

48K SPECTRUM FRED

Author: **Indescomp** ☐

Action beneath the Pyramids!

Fearless Fred the Intrepid Archaeologist searches the creepy catacombs below the torrid tomb of 'Tootiecarmoon' for the terrible treasures amidst monstrous mummies, ghastly ghosts, bats and rats!

COMMODORE 64 BOOGABOO (THE FLEA)

Author: **Indescomp** ☐

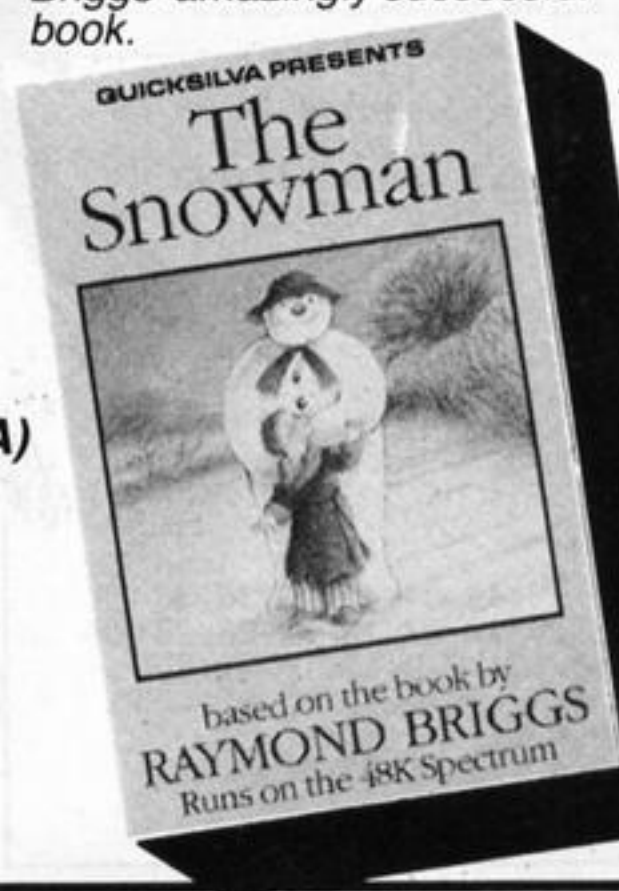
Itchy action!

Jump your way out of the caves with Bugaboo the flea but beware of the fearsome Dragon as you jump around the exotic vegetation.

48K SPECTRUM RAYMOND BRIGGS' THE SNOWMAN

Author: **David Shea**

An enchanting game based around episodes of Raymond Briggs' amazingly successful book.



All games marketed exclusively by Quicksilva Limited. Please send me the games I have ticked.

I enclose cheque/P.O. for _____
Send to Quicksilva Mail Order,
P.O. Box 6,
Wimborne, Dorset BA21 7PY.
Telephone: (0202) 891744

Name _____

Address _____



QUICKSILVA

WARNING: These programs are sold according to QUICKSILVA Ltd's terms of trade and conditions of sale. Copies of which are available on request.

YOU WON'T STOP PLAYING OUR TI-99/4A GAMES

LIONEL AND THE LADDERS



Fully Animated
Sprite Graphics
Multi-Screens
Hall of Fame

Willie, Horace, Cuthbert.
If you own a TI-99/4A
you missed a lot of FUN
Don't worry
LIONEL will change that

£7.95
EXT. BASIC

ADVENTUREMANIA

£5.95
A fascinating role playing graphic adventure game set in London for the Basic 16K TI-99/4A computer.

a graphic and text adventure



MANIA

BASIC £5.95

ADVENTUREMANIA 4 star review in H.C.W. Vol 45
The scene opens in Trafalgar Square. Your search for the hidden instructions will take you through the streets and buildings of London.
The screen displays: 3D View, Aerial View, Energy Level, Score etc., All at the same time.
Special routines include gambling casino, in game computer. Battle your wits against the guards but don't get caught. £5.95

MANIA 5 star review H.C.W. Vol 53
The quest continues:— You find yourself trapped in a chamber from which there seems to be no escape, suddenly you discover the key and your journey to MANIA has begun. How will you manage to survive in such strange lands as you travel through searching for the meaning of the ultimate quest and perhaps the corridors of time.
The screen displays: Your Score, Wealth, I.Q. and strength together with a graphic representation of your position and surrounding objects. There are 17 commands. The say command allows you to question the characters. £5.95

NO COMMAND MODULE IS NEEDED FOR THESE ADVENTURES

LIONEL AND THE LADDERS Ext Basic (New release)
The Sons of Mars have captured the beautiful Princess from Andromeda and carried her far away hiding her deep within a Labyrinth. Lionel sets out on the impossible mission to rescue her. The Sons of Mars move in to attack. Lionel runs through the Labyrinth climbing from one level to another he finds a key, unlocks a door and goes deeper into the Labyrinth where he encounters the Shades of Wrath and the Death Stars before he sets eye on his Princess. Can you help him succeed.
Fully animated sprite graphics etc. £7.95

Each game is packaged in an audio wallet.

A SELECTION OF INTRIGUE SOFTWARE STOCKISTS:

Blue Chip Computers
16 Clappgate Lane, Goose Green,
Wigan, WN3 6RN
Access & Barclaycard Tel: (0942) 322543

Estuary Home Computer Centre
261 Victoria Avenue, Southend-On-Sea, Essex
Access & Barclaycard Tel: (0702) 43568

P. V. Computers
38a Water Street, Accrington, Lancashire
Access & Barclaycard Tel: (0254) 36521/32611

Galaxy
60 High Street, Maidstone, Kent
Access & Barclaycard
Tel: (0622) 682575/679265

D. J. Software
Rosanar, The Wells, North Anston,
Sheffield, S31 7ED
Access & Barclaycard Tel: (0909) 562349

Software City
382 Kings Road, London, SW3 5UJ
Access & Barclaycard Tel: (01) 352 9220

Computer Corner
261, Commercial Road,
Portsmouth, Hampshire
Access & Barclaycard Tel: (0705) 833938

Computer Corner
5 Market Way, Havant, Hampshire
Access & Barclaycard Tel: (0705) 478572

Inkey Computer Services
70 Mill Street, The Square, Trecynon,
Aberdare, Wales
Access & Barclaycard Tel: (0685) 881828



Send PO/Cheque to:
INTRIGUE SOFTWARE
Cranbrook Road, Tenterden, Kent TN30 6UJ

INTRIGUE SOFTWARE
Tel: 05806 4726

Games to remember

SEND S.A.E FOR OUR SUPER GAMES LIST

Weedkiller

Your job is to destroy the weeds by moving over them. But don't run over a flower or the trail of poison you leave behind or you will lose one of your three lives.

With every move another flower appears and you'll have to be careful that they do not block your path before you run out of time, set at the beginning of the game.

If you clear all the weeds you move onto the next stage where there are more weeds, less time and more flowers to avoid.

Use the arrow keys to move your man.

Destroy the weeds... and try some sums

A game and a test for youngsters written for the unexpanded TI-994A by Cara and Paul Leathley

How it works

10-70 clear screen, display title
80-90 input time limit
120-170 define characters, colours.
180 set screen white
190-220 set variables
230-410 position weeds, flowers and man
420-430 check time not exceeded
440-810 detect keypresses, print new positions, leave trail
820-880 print new flower after each move
890 check man does not leave screen
920-930 checks for hit on flower, trail, weed
940-960 all weeds cleared?
970-1010 deduct one life, make sound
1020-1060 prepare more weeds, flowers for next stage
1070-1130 time up routine
1140-1220 lost three lives routine, another game?

```

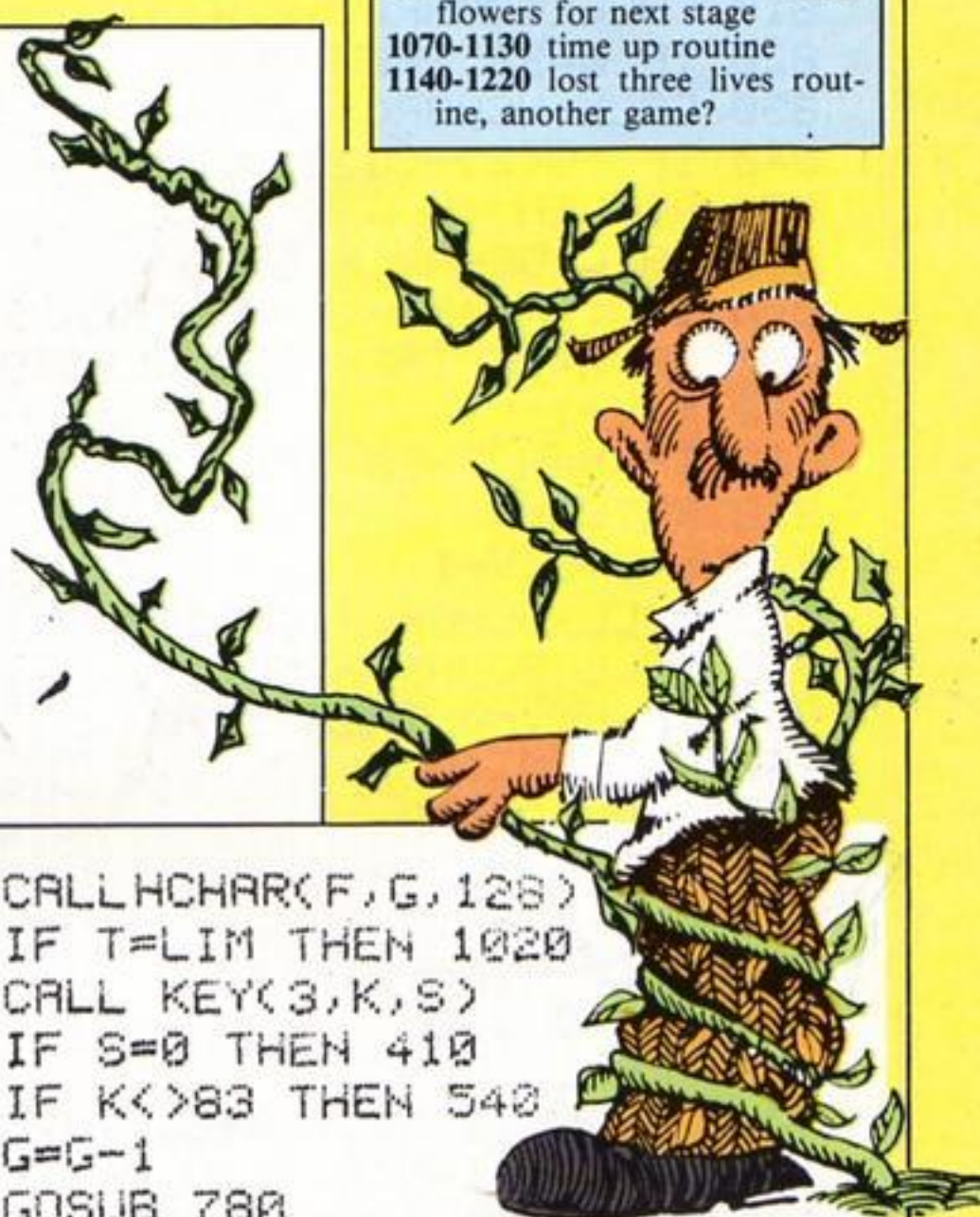
10 CALL CLEAR
20 PRINT TAB(9); "*****"
30 PRINT TAB(9); "* WEED KILLER *"
40 PRINT TAB(9); "*****"
50 PRINT
60 FOR D=1 TO 750
70 NEXT D
80 CALL CLEAR
90 INPUT "TIME LIMIT ?" : LIM
100 LIV=3
110 CALL CLEAR
120 CALL CHAR(96, "18107C54107C5410")
130 CALL CHAR(128, "183CFF183C246666")
140 CALL CHAR(145, "183C7E3C1D492A1C")
150 CALL COLOR(9,4,16)
160 CALL COLOR(13,9,16)
170 CALL COLOR(15,14,16)
180 CALL SCREEN(16)
190 N=60
200 H=25
210 M=0
220 T=0
230 FOR I=1 TO H
240 RANDOMIZE
250 A=INT(RND*22)+2
260 B=INT(RND*30)+2
270 IF (A=22)*(B=30) THEN 240
280 CALL HCHAR(A,B,96)
290 NEXT I
300 FOR J=1 TO N
310 RANDOMIZE
320 C=INT(RND*22)+2
330 D=INT(RND*30)+2
340 IF (C=22)*(D=30) THEN 310
350 CALL GCHAR(C,D,E)
360 IF E=96 THEN 310
370 CALL HCHAR(C,D,145)
380 NEXT J
390 F=22
400 G=30
410 T=T+1

```

```

420 CALL HCHAR(F,G,128)
430 IF T=LIM THEN 1020
440 CALL KEY(3,K,S)
450 IF S=0 THEN 410
460 IF K<>83 THEN 540
470 G=G-1
480 GOSUB 780
490 IF G<2 THEN 500 ELSE 520
500 G=2
510 GOTO 410
520 CALL HCHAR(F,G+1,46)
530 GOTO 410
540 IF K<>68 THEN 620
550 G=G+1
560 GOSUB 780
570 IF G>31 THEN 580 ELSE 600
580 G=31
590 GOTO 410
600 CALL HCHAR(F,G-1,46)
610 GOTO 410
620 IF K<>69 THEN 700
630 F=F-1
640 GOSUB 780
650 IF F<2 THEN 660 ELSE 680
660 F=2
670 GOTO 410

```



TI-994A PROGRAMS

Eric Adds Up

Designed to test addition and subtraction skills for children of pre-school age and above, the range of numbers in this program may be changed by altering the values of A and B in lines 430 and 440.

First the child's name is entered and then the choice is for addition or subtraction sums.

The sum is repeated 10 times and a score out of 10 is displayed.

How it works

70-90 assign light blue colour to character sets 2-12
100 GOTO instruction subroutine
110-170 define characters
190-410 clear screen, print Eric
420-440 assign random values to A and B
450-500 check for addition/subtraction, GOTO appropriate line
510-540 print addition sum
550-560 check for correct/incorrect answer
590-730 correct answer routine
740-900 incorrect answer routine
910-950 PRINT AT subroutine
960-970 input name
990-1200 clear screen, instructions
1210-1340 choose addition, subtraction
1350-1390 subtraction sum subroutine
1400-1490 end of 10 question subroutine, option for new test

Hints on conversion

TI BASIC is fairly standard but has some unique graphics commands, some of which are detailed below. The screen is made up of 32 columns and 24 rows.

CALL CLEAR clears screen
Replace with CLS or similar on other machines

CALL CHAR re-defines an ASCII character with a sixteen digit hex string derived from an 8 x 8 grid

CALL HCHAR (R,C,ASCII) places a character at row R and column C on the screen and optionally repeats it horizontally

CALL VCHAR similar to HCHAR but repeats character vertically

CALL GCHAR (R,C,X) repeats the ASCII code to variable X of screen position R and C. Similar to a screen PEEK on other machines

CALL KEY (3,K,S) places the code of the key predded into the variable K. Similar to GET on other computers

```

680 CALL HCHAR(F+1,G,46)
690 GOTO 410
700 IF K<>88 THEN 410
710 F=F+1
720 GOSUB 780
730 IF F>23 THEN 740 ELSE 760
740 F=23
750 GOTO 410
760 CALL HCHAR(F-1,G,46)
770 GOTO 410
780 RANDOMIZE
790 Y=INT(RND*22)+2
800 Z=INT(RND*30)+2
810 CALL GCHAR(Y,Z,W)
820 IF W=96 THEN 780
830 CALL HCHAR(Y,Z,145)
840 IF (G<2)+(G>31)+(F<2)+(F>23) THEN 850 ELSE 860
850 RETURN
860 CALL GCHAR(F,G,X)
870 IF (X=145)+(X=46) THEN 920
880 IF X=96 THEN 890 ELSE 910
890 M=M+1
900 IF M=H THEN 970
910 RETURN
920 LIV=LIV-1
930 CALL HCHAR(F,G,128)
940 CALL SOUND(-750,110,0,111,0,112,0)
950 IF LIV=0 THEN 1090
960 RETURN
970 CALL CLEAR
980 H=H+5
990 N=N+5
1000 LIM=LIM-(LIM/10)
1010 GOTO 210
1020 CALL CLEAR
1030 CALL SOUND(-1000,110,0,111,0,112,0)
1040 M$="T I M E      U P      !"
1050 FOR I=1 TO LEN(M$)
1060 CALL HCHAR(11,I+6,ASC(SEG$(M$,I,1)))
1070 NEXT I
1080 GOTO 1140
1090 CALL CLEAR
1100 M$="YOU LOST YOUR 3 LIVES"
1110 FOR I=1 TO LEN(M$)
1120 CALL HCHAR(11,I+5,ASC(SEG$(M$,I,1)))
1130 NEXT I
1140 PRINT "ANOTHER GAME? PRESS 'Y'"
1150 CALL KEY(3,K,S)
1160 IF S=0 THEN 1150
1170 IF K=89 THEN 60 ELSE 1180
1180 END

```



TI-994A PROGRAMS

```

10 REM *****
20 REM * ERIC ADDS UP *
30 REM *****
40 REM
50 REM BY PAUL LEATHLEY
60 CALL SCREEN(16)
70 FOR C=2 TO 12
80 CALL COLOR(C,5,16)
90 NEXT C
100 GOSUB 960
110 CALL CHAR(97,"1010101010101010")
120 CALL CHAR(98,"000000FF")
130 CALL CHAR(99,"00000000000000FF")
140 CALL CHAR(128,"1028444444443800")
150 CALL CHAR(100,"0000824438")
160 CALL CHAR(130,"0102040810")
170 CALL CHAR(131,"8040201008")
180 FOR J=1 TO 10
190 CALL CLEAR
200 PRINT " *****"
210 PRINT " bbbbbbbbbb"
220 PRINT "*****"
230 PRINT " a a "
240 PRINT " a ~ ~ a "
250 PRINT " a @ @ a "
260 PRINT " a a "
270 PRINT " a / a "
280 PRINT " a a "
290 PRINT " a cccc a "
300 PRINT " a a "
310 PRINT " \cccc/ "
320 PRINT " a a "
330 PRINT " /bbbbbb\ "
340 PRINT " a/bbbb\ a "
350 PRINT " a.a I A m.a "
360 PRINT " aa aa "
370 PRINT " \ERIC/ "
380 PRINT " (<\bb//> "
390 PRINT " aa^ ^aa "
400 PRINT " aad daa "
410 PRINT " ccc ccc"
420 RANDOMIZE
430 A=INT(RND*10)
440 B=INT(RND*10)
450 IF (P=2)*(A<B) THEN 420
460 IF P=1 THEN 470 ELSE 490
470 G=A+B
480 GOTO 510
490 G=A-B
500 GOTO 1350
510 C=19
520 R=9
530 M$=STR$(A)&" + "&STR$(B)&" ="
540 GOSUB 910
550 INPUT T
560 IF T=G THEN 590 ELSE 740
570 NEXT J
580 GOTO 1400
590 CALL HCHAR(10,6,92)
600 CALL HCHAR(10,11,47)
610 CALL HCHAR(5,7,32,4)
620 CALL HCHAR(4,7,126)
630 CALL HCHAR(4,10,126)
640 CALL SOUND(100,587,1,880,1,740,1)
650 CALL SOUND(1000,587,1,880,1,740,1)
660 C=14
670 R=15
680 M$="WELL DONE "&N$
690 GOSUB 910

```

```

700 SC=SC+1
710 FOR D=1 TO 750
720 NEXT D
730 GOTO 570
740 CALL HCHAR(11,6,130)
750 CALL HCHAR(11,11,131)
760 CALL HCHAR(6,7,100)
770 CALL HCHAR(6,10,100)
780 CALL HCHAR(8,10,128)
790 CALL SOUND(-1000,110,0,111,0,112,0)
800 C=14
810 R=15
820 M$="HARD LUCK "&N$
830 GOSUB 910
840 C=14
850 R=18
860 M$="THE ANSWER IS "&STR$(G)
870 GOSUB 910
880 FOR D=1 TO 750
890 NEXT D
900 GOTO 570
910 FOR I=1 TO LEN(M$)
920 CALL HCHAR(R,C+I,30)
930 CALL HCHAR(R,C+I,ASC(SEG$(M$,I,1)))
940 NEXT I
950 RETURN
960 CALL CLEAR
970 INPUT "WHAT IS YOUR NAME?":N$
980 GOSUB 1210
990 CALL CLEAR
1000 PRINT "HI ",N$
1010 PRINT "*****"
1020 PRINT
1030 PRINT "SOON YOU WILL MEET ERIC."
1040 PRINT
1050 PRINT "ERIC IS NOT TOO GOOD AT"
1060 PRINT
1070 PRINT "SUMS SO YOU WILL HAVE TO"
1080 PRINT
1090 PRINT "HELP HIM. IF YOU GET THE"
1100 PRINT
1110 PRINT "SUM RIGHT HE WILL SMILE"
1120 PRINT
1130 PRINT "BUT IF YOU ARE WRONG HE"
1140 PRINT
1150 PRINT "WILL START TO CRY."
1160 PRINT
1170 PRINT "PRESS ANY KEY TO START"
1180 CALL KEY(0,K,S)

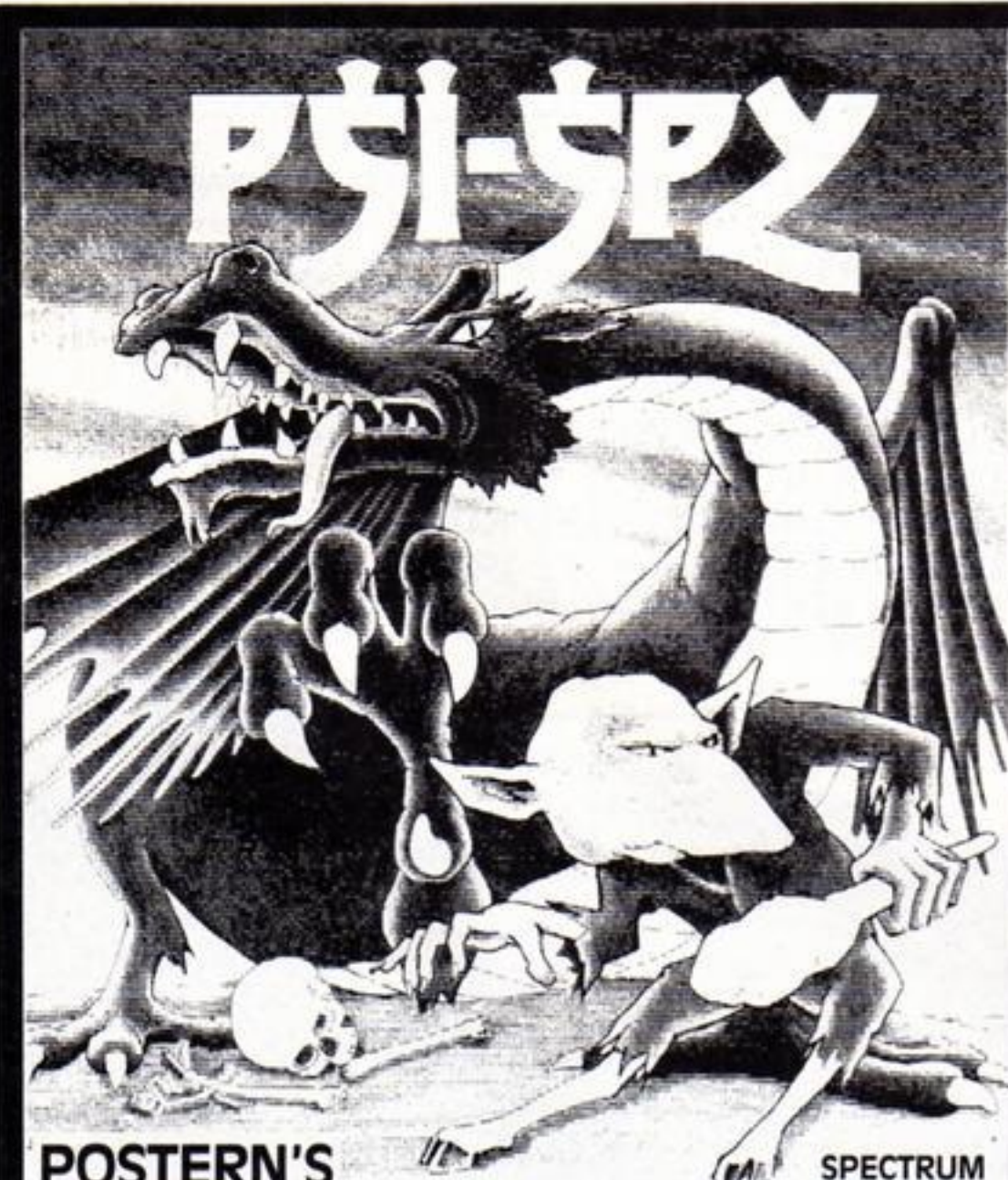
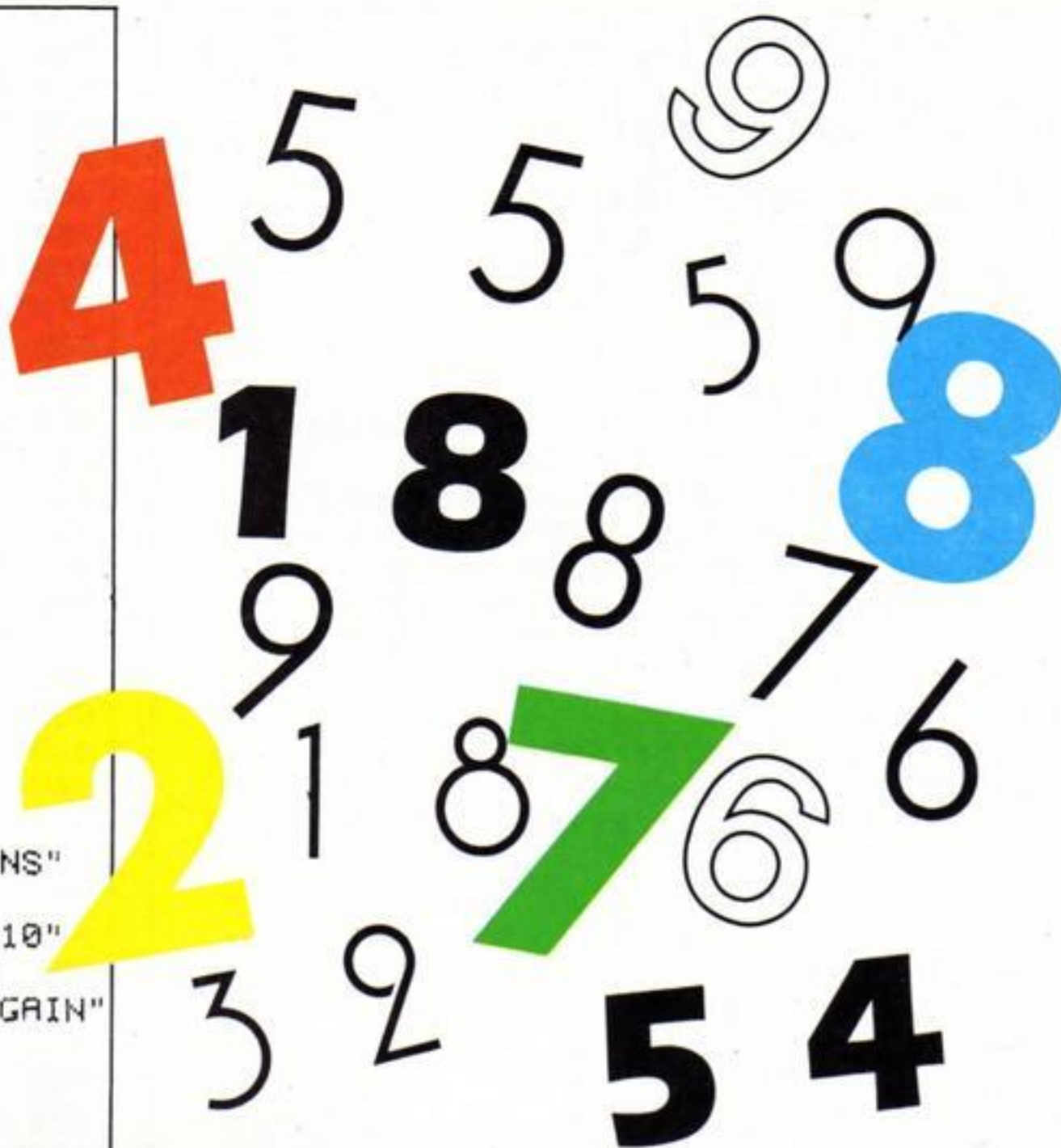
```



TI-994A PROGRAMS

```

1190 IF S=0 THEN 1180
1200 RETURN
1210 PRINT
1220 PRINT "WHAT DO YOU WANT TO DO?"
1230 PRINT
1240 PRINT "PRESS : "
1250 PRINT
1260 PRINT "1 TO DO ADDS"
1270 PRINT "2 TO DO TAKE AWAYS"
1280 CALL KEY(0,K,S)
1290 IF K=49 THEN 1310
1300 IF K=50 THEN 1330 ELSE 1280
1310 P=1
1320 RETURN
1330 P=2
1340 RETURN
1350 C=19
1360 R=9
1370 M$=STR$(A)&" - "&STR$(B)&" ="
1380 GOSUB 910
1390 GOTO 550
1400 CALL CLEAR
1410 PRINT "YOU'VE DONE YOUR QUESTIONS"
1420 PRINT
1430 PRINT "AND SCORED",SC," OUT OF 10"
1440 PRINT
1450 PRINT "PRESS ANY KEY TO START AGAIN"
1460 CALL KEY(0,K,S)
1470 IF S=0 THEN 1460 ELSE 1480
1480 SC=0
1490 GOTO 10
    
```



POSTERN'S LATEST CHALLENGE.

Combining the best of two alien worlds on one mind-blowing cassette.

£7.95 WITH KURRAH SPEECH

Postern Ltd, PO Box 2, Andoversford, Cheltenham, Gloucestershire GL54 5SW

SPECTRUM 48K



POSTERN

FOR ALL THE FAMILY



Games from the Gods

EDUCATIONAL

FOR UPTO 4 PLAYERS



FOR UPTO 4 PLAYERS

QUIZ

GEOGRAPHY

FOR ANY 48K ZX SPECTRUM

£5.95

DISTRIBUTORS / DEALERS ENQUIRIES WELCOME

TELEPHONE: 051-263-8521/2

Jogging/ Sidewinder Sord M5 £8.95

Computer Games, Goldings Hill,
Loughton, Essex IG10 2RR

In the first game you control John, who goes jogging every morning. He wants to win the big race next week. A rival, however, is trying to interfere. He digs holes in the road, drops apples in his path, and uncages a very unfriendly crow.

You will be pleasantly surprised when you run this game. John is a brightly coloured, well-defined character and as he waves his arms and legs, background scrolls to give the impression of movement. You can make him run, stop and, somehow, go up and down!

In Sidewinder, half of the screen is the sky and bottom half is the instrument panel. The sky has dozens of dots — enemy helicopters, jet planes, prop planes. You have to destroy as many of them as you can using your sidewinder missiles before your fuel runs out. Of course, before this happens you must land.

Although both these are written in BASIC, they feature good graphics and sound. They're obviously not as sophisticated as cartridge games, but represent good value for money. S.S.

instructions	100%
playability	70%
graphics	70%
value for money	80%



Ghouls 32K BBC £7.95

Micro Power, 8/8a Regent St,
Chapel Allerton, Leeds LS7 4PE

Another high standard Program Power game. Cassette presentation is very good and instructions are quite adequate. As the game loads we can hear spooky music

and the screen display changes colour, making the long loading much more interesting.

Starting at the bottom, the idea is to reach the top and enter the next room of the creepy mansion. During your trip you have to eat titbits to gain extra points, avoid poison-smeared spikes and bouncing spiders. Your journey is made more difficult by moving platforms,

contracting floorboards and powerful springs. If you think that sounds easy, there's a time limit and, if you fail to reach the top, you shrivel up into a little pile.

Graphics are very good indeed. The slightest things have been thought of. When you die, the ghost who chases you around the screen smiles. Four levels have to be accomplished before you gain an extra life. The first three are deadly, but the

fourth... who knows?

Addictive and exciting. An interesting feature is that when you lose a life the screen moves away sideways and reappears in a spin to the centre. D.B.

instructions	85%
playability	90%
graphics	95%
value for money	90%



Jet Power Jack 32K BBC £7.95

Micro Power, 8/8a Regent St,
Chapel Allerton, Leeds LS7 4PE

Jack has been captured by the evil warlord Nogrob. You have to collect fuel pods from one side of the garage and return to the other side to refuel the waiting starship. Just to add excitement, Jack has to avoid nasty space Googlies and finally, yes, you've guessed, prepare yourself, it's... Wilfred the hideous Yugg monster. To add to this impossible task, the sides of each ledge are live. Touching them will cause Jack to plummet to his death.

Is this game difficult? You're not kidding. After plenty of practice I finally got the hang of moving Jack around. Now, I'm

starting to play the game. If you like challenges you'll like this. Presentation is of the high standard we have come to expect from Program Power. However, you must prepare yourself for the pictures of the terrible alien creatures, such as Wilfred.

This game takes a lot of getting used to, but it's worth the trouble. Possibly not very original, I think new ideas are getting few and far between. Still, I'm not trying to put anyone off. D.B.

instructions	90%
playability	85%
graphics	90%
value for money	95%



3-D Stalkers TI-99/4A £5

Stainless, 10 Alstone Rd, Stockport, Cheshire

An interesting tactical game which requires two players. You are both trapped in a symmetrical maze from which only one of you may escape. The object is to pursue and kill your opponent before he finds and eliminates you.

The screen shows an overhead view of the maze and 3D perspectives of what is directly ahead of each player, updated after each move. Your position in the maze is not shown, but may be displayed at any time if requested. Obviously this option should not be used arbitrarily. Confirmation of your whereabouts is not only critical to you, but of certain interest to your opponent.

The length and difficulty of each game is determined by the number of shots that are required to kill your opponent, requested before each game starts. Movement, controlled by keyboard or joystick, is forward, or turn back, left or right.

Not the sort of game I find particularly addictive, but at least it does make a change to pitch your wits against an equally fallable human opponent \$ and who knows, will you be the hunter or the hunted? J.W.

instructions	90%
playability	60%
graphics	50%
value for money	85%



Raging River/ Super Jackpot TI-99/4A £6

Pete Williams, from Stainless, 10
Alstone Rd, Stockport, Cheshire

The first is actually called The Raging Amazon, and one wonders why the author gave it two titles. It is of the ski-run type: you must glide your canoe through a vertically-moving

panorama. Graphics are very good, and good use has been made of colour. Hazards include fallen trees lying half-submerged, rocks and man-eating alligators (who also have a penchant for canoes it seems). You can speed up or slow down the movement by controlling the rate at which the panorama moves past you. There are occasionally delays while a melody is played — at the beginning and whenever you fall

foul of a hazard. The cassette label claims that you should aim to travel 82 miles — it can be done, it says — but I found that frequently the river was blocked completely by more than one hazard and it was quite an effort to travel just a few miles. The second is yet another in the fruit machine mould and, although the graphics are excellent and every effort has been made to overcome the slowness of TI BASIC, in the end it is just as

boring as all the others on the market. P.B.

instructions	80%
playability	75%
graphics	90%
value for money	65%



Texas: the state of the stars

A mixture of five new games for the computer that won't lie down

Pelmanism £3.95

MarKat, 4 Cloreen Park, Belfast BT9 5BX

An entertaining game for adults and children alike which relies on players being able to remember a set of 32 cards displayed face-down.

Player one starts by selecting one of the cards by pressing A to Z and 0 to 9. The chosen card is then displayed face-up and the player makes another choice. If the two cards match, they are turned over and play passes to the opponent.

Obviously, play becomes easier as play progresses, because the identity of each card is revealed and there are less to remember.

The pictures used on the cards are four characters in size and include houses, trees, flowers, faces and patterns.

Unfortunately, this is for two players only. An option to have four or five competing would be nice, as would an option to compete against the computer. This would obviously have to rely on a rather unsatisfactory random method, determining whether or not the computer memorised the cards correctly.

J.J.

instructions	75%
playability	70%
graphics	70%
value for money	70%



Hangmog £3.20

SoftI, 14 Station Rd, Brough, North Humberside

A clever and original idea based on Hangman. The prisoner on the gallows is replaced with a rather unfortunate cat which has strayed into the meat works and is caught on a conveyor belt leading to a mincer.

Your job is to guess a word and save moggy.

The alphabet is displayed at the top with a conveyor belt underneath, onto which the letters fall once chosen.

Depending on whether or not the letter chosen is correct, it falls either down the "best beef" chute or lands on the conveyor belt on which moggy is seated.

When a wrong letter falls onto moggy's conveyor belt it pushes a chunk of meat off the other end and into the mincer. As cats have nine lives, moggy has eight chunks of meat in front of him and on the ninth incorrect guess he runs out of luck and falls down the mincing chute.

Unfortunately, it is possible to press the same letter any number of times, although, if the player is careless enough to press the same letter twice, he deserves to lose a life.

Overall, well-thought and enjoyable.

J.J.

instructions	70%
playability	80%
graphics	95%
value for money	85%



ASW Tactics 2 £11

DEJ, from Stainless, 10 Alstone Rd, Stockport, Cheshire

After first reading the instructions, full of high powered jargon, I was filled with enthusiasm. However, apart from some original graphic effects and sounds, it fell far short of my expectations.

A U.S. naval officer wrote this and it is a simulation based on his experience as a helicopter pilot.

The screen shows an aerial view of a task force, led by a carrier for the helicopter. You, the 'copter pilot, must safeguard the fleet while searching for an enemy submarine.

To gain information on the

sub's location, the helicopter flies around dropping sonar buoys, using onboard sonar and, if requested, looking for a signal from the surveillance S-3 jet.

Providing a further clue you may request a three-second periscope view of the task force, as seen from the enemy submarine. I failed to believe the accuracy of this when, only having two ships left in the fleet, the periscope still showed three!

After destroying the sub you are greeted by the Stars and Stripes — actually the Squares and Stripes — and a mini firework display. Needs Extended BASIC.

J.W.

instructions	100%
playability	35%
graphics	60%
value for money	45%



Pilot £5.95

Apex, 115 Crescent Drive South, Brighton

A good well-written flight simulator. Screen display is clear and the colours are well contrasted. The display of the terrain map is blocky, but the runway display is good, shown from above as your plane slowly approaches.

The tape loaded first time to give the option of take off, landing or in flight. You are then asked if you require wind effects and the game starts.

In play the program is slow, but gives a reassuring beep when you press any key. There is an audible stall warning, otherwise all is quiet. The instructions

come in a four-page booklet as well as a separate sheet giving loading instructions. The instructions are easy to understand and are clearly set out, like the whole program. The program is easy to use, the slowness being to the players advantage.

Overall, fun and very addictive at the start. Later on, however, the program did become a bit boring, after I had mastered landing. Everything is taken into consideration, such as wind, lift and flap drag.

M.H.

instructions	100%
playability	85%
graphics	80%
value for money	95%



Go for the gold £7

DEJ, from Stainless, 10 Alstone Rd, Stockport, Cheshire

If you've ever fancied searching for the little people's pot of gold then there's plenty here. They're also so easy to obtain that you'll be bored stiff collecting them.

The idea is that you are lost in various forests looking for the gold. The screen shows trees, your man and a pot of gold. Also in the forest are pixies, to help increase your speed, but in pursuit of your man there's a leprechaun. When he catches you a man is lost. The game continues for as long as you have men. With each pot collected its

value increases and you move onto the next forest.

Designed for keyboard or joystick, in the keyboard version it seems a gremlin has joined forces with the little people! by pressing any key other than the arrow keys used in the game your man travels around and off the screen at top speed. With this method of cheating, pots of gold become yours for the taking. At least this provides some amusement in what is otherwise a slow and tedious game. Needs Extended BASIC.

J.W.

instructions	50%
playability	20%
graphics	40%
value for money	25%



Guttong Gotton Sord M5 £23.95

Computer Games, Goldings Hill, Loughton, Essex IG10 2RR

You play the part of a railroad builder. First you have to lay tracks to the station and then drive the train through the stations.

The screen is divided into 24 blocks, one of which is empty, exactly like sliding block puzzles, popular some years back. Each block has a track on it and you have to move them so that the train, which is moving continuously, won't go into a dead end. It's no good simply keeping the train running as you have to collect all the passengers at the stations, located at the

edge of the screen, to go on to the next screen. There are enemy trains which try to crash into you but, but by manipulating the tracks, you can make them crash into each other.

The graphics are very nice and sounds very impressive. The game has some jolly tunes and the overall presentation is excellent. If you want a mind stimulating game which also has the addictive quality of a real time arcade game, buy this. It is great fun to play and is refreshingly different. S.S.

instructions	90%
playability	90%
graphics	90%
value for money	90%



Step Up Sord M5 £23.95

Computer Games, Goldings Hill, Loughton, Essex IG10 2RR

The instructions explain that you have to climb 10 flights of stairs, jump over the obstacles or evade them with your barrier, get to the top and escape with the friendly UFO before your energy runs out. You control a cute little figure and make him climb stairs. With a quick push on the left edge of the attack button, you can make him jump, or the right edge to make him invincible for a few seconds.

Even on the first screen, there is a huge number of nasties, including spiders, mice, spider's webs and monsters wandering around and, at first, it can be

frustrating as you lose your three lives in a matter of seconds. But if you persevere, you will eventually get to the top to get rescued by the UFO, then there's the next screen where there's even more spiders, mice, monsters...

The characters are well defined and animation is good. However, the sprite feature is not used to its fullest. In fact, some characters aren't even sprites, but hardened arcade players could do worse than buying this highly original game. S.S.

instructions	80%
playability	70%
graphics	70%
value for money	80%



Backgammon TI-99/4A £7

Not Polyoptics, from Stainless, 10 Alstone Rd, Stockport, Cheshire

This opportunity to challenge the TI was greeted enthusiastically by our household, but we were soon to be disappointed.

The program contains options to either play against the computer or another player. During the first attempt at the one player game, while the computer was considering its second move, the program crashed with a Bad Value in 2010.

The second game I managed to win, but at its end the board disappeared and I was informed my score was now one, meaning I had won one game. Scoring in the original game is done on a points system, the most being awarded for a backgammon. Omitting this seems very strange, particularly as the program includes an option to double.

On my third attempt everything was just going well when the display partially scrolled off the screen and the program locked. This couldn't be cleared, so I had to quit and re-load.

Fortunately, several subsequent games passed uneventfully, but the computer proved to be a fairly pedestrian and not altogether intelligent opponent. In view of the limited memory available, the program overall seems reasonable, but has little to offer an experienced player. Needs Extended BASIC. J.W.

instructions	60%
playability	50%
graphics	50%
value for money	80%



For the fun of it...

Build a railway, play
backgammon... two games in
this collection tested by our
review panel

Snakey/ Barrier Attack Sord M5 £8.95

Computer Games, Goldings Hill, Loughton, Essex IG10 2RR

Snakey is a typical snake game where you control a hungry snake that feeds on frogs whilst avoiding rocks and the electrified fence. Also, as Snakey grows he restricts his own movement. The snake is very colourful, movement is fast and key response is good. In the tiny memory of the standard M5, it even has a joystick option and difficulty levels.

The second game is rather disappointing in comparison. The blurb says that in the year 4501, the uneasy cosmic peace is about to be shattered by barbarians, the uncouth warriors

of the dreaded Oasys Squadron. You, as the commander of the 17th zone Defence Battalion, must prevent them from destroying your ship and over-running your section of the galaxy, and so on. Sounds incredibly exciting doesn't it? Well, it's a simple vertical scrolling game. Your ship at the top of the screen can be moved left and right. By pressing 1 or the attack button, you can clear a path. Graphics are simple and sound is boring.

To conclude, not a bad package, second game could have been better. S.S.

instructions	95%
playability	60%
graphics	60%
value for money	70%



Quasimodo/ Q Bono TI-99/4A £8

Stainless, 10 Alstone Rd, Stockport, Cheshire

Here's two Extended BASIC games for the high scorers among us. Each is loaded separately. Quasimodo requires the use of a joystick, but Q is for keyboard only.

In Quasimodo you have five lives with which to survive as many screens as you can — ringing bells all the way! The purpose is to rescue damsels in distress. On each of several screens you run and jump along a brick wall. Hazards to avoid are combinations of arrows, cannonballs and holes in the wall. Points are scored for ringing a bell or rescuing a maiden.

Q Bono is a simple yet addictive game. Your man starts at the top of a pyramid of cubes. Using four arrow keys to control his diagonal movements he must travel across all the cubes which changes their colour to cyan. He is pursued by a snake which will take one of his lives if both land on the same cube together. After the first screen each cube must be visited twice. In subsequent screens this number increases. If you've a black and white TV, numbers can be shown on the cubes to indicate colour changes. J.W.

instructions	80%
playability	80%
graphics	75%
value for money	100%



Dragon Cruncher Dragon 32/ Tandy Colour £7.95

Elkan Electronics, 11 Bury New Rd, Prestwich, Manchester M25 8JZ

This has got to be a mutually beneficial program for these two computers. They use the same 6809 chip and this makes it easy to convert BASIC programs from one to the other, which it does quickly and efficiently. There are only a few reservations, one being that it will ignore 'machine code and the contents of REM and STRINGS'.

It will allow any number of programs to be converted on one loading so long as you do not RUN the program or PCLEAR 1.

There are certainly many programs for each of the computers which can now be converted and so extend the range of available software.

A simple and effective utility that may prove a saviour for the Tandy owner who feels increasingly isolated. **M.P.**

instructions	100%
ease of use	100%
display	N/A
value for money	80%



Minitext Dragon 32 £9.95

Microplot, 19 The Earls Croft, Cheylesmore, Coventry CV3 5ES

This is an interesting utility as it has a dual purpose to fulfil as both a text editor (or small word processor) and BASIC program manipulator. It has plenty of commands, in fact probably too many to remember. They are mainly activated by pressing the 'clear' key followed by an alphabetic key with some sort of relationship to the command desired. So, for example, C centres text, L loads or joins a file to the one currently in memory and J justifies lines.

There is a command for every letter key but some are unlikely

to be used regularly and are more difficult to learn. The cursor control keys have several functions including underline on the shift/cursor up which is very simple.

The problem with editing BASIC programs is that you cannot test your changes without first saving, but this is to be expected from a word processor. Another thing that is difficult to get used to is that you don't need to press 'enter' after typing a line. If you do accidentally press that key during program editing real problems can result. **D.C.**

instructions	70%
ease of use	80%
value for money	70%



Editor/Reseq & Renum Dragon 32 £7.95/£5.95

Microplot, 19 The Earls Croft, Cheylesmore, Coventry CV3 5ES

The reason for reviewing these two programs together is that Reseq/Renum is simply a subset of Editor. These are BASIC programming utilities and allow you to manipulate programs you have recorded on tape using the CSAVE command.

The 'Reseq' command allows you to move chunks of program to a new position, valuable if you like to develop programs in small chunks ready for final assembly. 'Renum' can renumber selected program lines.

The full editor has further commands, the most useful of which is 'Merge' which will add and sort another program to the one already in memory. The 'Delete' command allows block deletion and a line joining facility helps to save memory. There is a global search and replace facility called 'Find' and a 'Copy' facility to save retyping commonly used lines.

Its biggest failing by far is that you cannot run the program whilst editing it. You must first save to cassette and then reload to test your alterations.

I cannot really see why anyone would buy the subset tape unless they need the extra memory it leaves (just over 1K). **D.C.**

instructions	80%
ease of use	70%
value for money	70%



Make more use of your micro

Useful utilities are given the
once-over by our panel of
experts

Autochar File TI-99/4A £4.95

Autochar File is not a game but a valuable programming aid for those who would like to devise their own games but are put off by the work involved in setting up graphics characters.

On loading this program a box is depicted on screen with a black square in it. By using the direction keys you can move around in the box and print the square by pressing the number one key. If an alteration is required a square may be deleted by pressing the number zero key.

In this way characters may be built up and when the desired format is achieved you simply press enter and the char code is

shown. A small diagram is also displayed showing the effect the character gives when printed across the entire screen.

Additional facilities are that char codes may be saved for use at a later date, enabling the user to build an entire library of character codes.

At £4.95 I think this program is invaluable to the programmer as a great effort- and time-saving device. **D.B.**

instructions	90%
ease of use	100%
display	80%
value for money	90%



Music Synthesizer TI-99/4A £10

Norton, from Stainless, 10, Alstone Rd, Stockport, Cheshire SK4 5AM

This program, also available in TI BASIC, is in two parts. The first, quite large, program is the synthesizer, and the second enables you to play the music created by the first and stored on another tape as a series of data files. I don't know how it compares with the TI module which performs a similar task, but as an amateur musician who can't read a note it held my interest and taught me a thing or two — not very difficult.

Having set up sharps and flats and chosen a tempo, you are presented with a fairly complex display consisting of the treble and bass staves: graphics representing musical notes and options for choosing one of three notes for the 'voice' with which you will 'draw' or play.

It is possible to create music using a score which you may have bought and to produce most necessary effects — you can even transpose a voice up or down. You can play all three voices or any one independently, and once edited to your satisfaction using the usual eight direction keys, you can store on tape.

There is more to this program, and it is a very useful utility for someone who wants to produce a melody but doesn't want the chore of having to learn music in any depth. **P.B.**

instructions	100%
ease of use	90%
display	95%
value for money	80%



Deal yourself in and risk the turn of the cards

Forget the cards — with Jon Dempsey's program for the unexpanded TI-994A it's just as easy to win (or lose) on the screen



Variables

H Points high score
Points points scored
BET bet
P,H ASCII values of cards
PS ASCII value of points for use in PRINT AT routine
AS string to be printed
RW row number
A ASCII value of string to be printed at
KK graphics characters for cards
Q column number for cards
PP,Y graphics characters
QW ASCII value of card position
X value returned by CAL GCHAR to check card
M,K card variables
QS string variables

In my program for the standard TI-99/4A you are in the position of one of the contestants in TV's Play Your Cards Right.

After the screen titles and rules have cleared, you are shown one of five cards. You must then bet points (from one to nine) on whether the next card will be higher or lower.

How it works

100-150 reset
180-350 redefine characters
360-410 set colours
420-810 print screen titles and instructions
860-940 high-score routine
1030-1790 main sequence
1850-2440 select and print cards
2470-2510 PRINT AT routine
2750-2970 check cards
3000-3270 end

Hints on conversion

TI BASIC is fairly standard with two main exceptions. Single statement lines are used and subroutines for graphics and sound are available, prefixed with CALL, as detailed below.

CALL CLEAR clears screen

CALL SCREEN defines screen colour

CALL CHAR defines a character with a 16-digit hex string from an eight by eight grid. Other machines will use a different method

CALL COLOUR (character set, foreground, background) sets colour of character

CALL HCHAR (row, column, character code, number of repeats) places character on screen at co-ordinates specified and optionally repeats character horizontally

CALL VCHAR similar to HCHAR but repeats character vertically

CALL GCHAR (row, column, variable) similar to a screen PEEK on other machines. Gives the ASCII code at the given screen co-ordinates

CALL SOUND (D,F,V) gives sound of duration D, frequency F and volume V. Three sounds can be produced at once

High score and highest scorer is stored and displayed on the screen.

An unusual routine in the program is the storing of values and strings for print at routines in DATA statements.

```
100 RESTORE
110 DIM H%(14)
120 DIM L%(14)
130 X%=H%:HI SCORE=
140 HPOINTS=25
150 RANDOMIZE
160 CALL CLEAR
170 CALL SCREEN(2)
180 CALL CHAR(128,"FFFFFFFFFFFFFF")
190 CALL CHAR(152,"0103070F1F3F7F7F")
200 CALL CHAR(127,"A55A55A55A55A5")
210 CALL CHAR(153,"B0C0E0F0F0FCFEFE")
220 CALL CHAR(154,"7F7F3F0F030307")
230 CALL CHAR(155,"FEF0FC0C0C0E0")
240 CALL CHAR(156,"01070707073B7FFF")
250 CALL CHAR(157,"B0E0E0E00C0CFE")
260 CALL CHAR(158,"FFFFFFF73B030307")
270 CALL CHAR(159,"FFFFFFFF0C0C0E0")
280 CALL CHAR(144,"0103070F1F3F7FFF")
290 CALL CHAR(145,"B0C0E0F0F0FCFEFE")
300 CALL CHAR(146,"FF7F3F1F070301")
310 CALL CHAR(147,"FF0FC0F0E0C030")
320 CALL CHAR(148,"001C3E7FFFFFFF")
330 CALL CHAR(149,"00367CFEFFFFFF")
340 CALL CHAR(150,"7F3F1F070301")
350 CALL CHAR(151,"FEF0FC0C0C0E0")
360 CALL COLOR(15,7,1)
370 CALL COLOR(16,5,1)
380 CALL COLOR(12,16,7)
390 FOR I=2 TO 8
400 CALL COLOR(1,14,1)
410 NEXT I
420 FOR I=1 TO 4
430 READ H%(I),X%(I),Z%(I)
440 H%=H%(I)
450 X%=X%(I)
460 F=Z%(I)
470 GOSUB 2470
480 NEXT I
490 DATA CARD SHARK,5,11,*****6,10, J.DEMPSEY ,0,6,PRESS ANY KEY,
TO PLAY,22,3
500 FOR RA=1 TO 4
510 IF RA<4 THEN 530
520 RESTORE 630
530 CALL KEY(0,K,S)
540 IF S<>0 THEN 700
550 READ T%(RA),QD(AR)
560 H%=T%(AR)
570 Y=QD(AR)
580 RW=15
590 F=12
600 GOSUB 650
610 GOSUB 2470
620 NEXT RA
630 DATA " SPADES ",152," CLUBS ",156," DIAMONDS",144," HEARTS ",148
640 GOTO 500
650 CALL HCHAR(12,16,Y)
660 CALL HCHAR(12,17,Y+1)
670 CALL HCHAR(13,16,Y+2)
680 CALL HCHAR(13,17,Y+3)
```

```
690 RETURN
700 CALL CLEAR
710 RESTORE 790
720 FOR III=1 TO 12
730 READ H%(III),L%(III)
740 H%=H%(III)
750 RW=III*2
760 F=L%(III)
770 GOSUB 2470
780 NEXT III
790 DATA *****10,RULES,11,*****10,THIS IS A GAME OF"PLAY YOUR,3,CARDS RIGH
T",10
800 DATA YOU GUESS WHETHER THE NEXT,3,CARD WILL BE HIGHER OR LOWER,3,THAN THE LA
ST CARD,3
810 DATA USE THE ARROW KEYS E AND X,3,TO SELECT VALUES,3,MORE INSTRUCTIONS,5,WIL
L APPEAR DURING PLAY,3
820 FOR D=1 TO 1000
830 NEXT D
840 CALL CLEAR
850 GOTO 950
860 CALL SOUND(200,370,2,440,2,294,2)
870 CALL SOUND(600,392,2,494,2,294,2)
880 CALL SOUND(200,391,2,494,2,292,2)
890 CALL SOUND(800,294,2,440,2,370,2)
900 INPUT "ENTER YOUR NAME "B%
910 IF LEN(B%)>16 THEN 840
920 CALL SCREEN(13)
930 CALL CLEAR
940 RETURN
950 CALL SCREEN(13)
960 FOR I=2 TO 8
970 CALL COLOR(1,2,16)
980 NEXT I
990 CALL COLOR(13,16,16)
1000 CALL COLOR(15,7,16)
1010 CALL COLOR(16,2,16)
1020 GOSUB 3180
1030 CALL CLEAR
1040 H%=X%:"&STR$(HPOINTS)
1050 RW=3
1060 F=3
1070 GOSUB 2470
1080 H%="POINTS="
1090 RW=20
1100 F=16
1110 GOSUB 2470
1120 GOSUB 2690
1130 FOR I=5 TO 25 STEP 5
1140 FOR D=10 TO 13
1150 CALL HCHAR(D,1,127)
1160 CALL HCHAR(D,1+1,127)
1170 NEXT D
1180 NEXT I
1190 FOR Q=5 TO 25 STEP 5
1200 K=INT(4*RN)+1
1210 ON K GOSUB 1880,1900,1920,1940
1220 M=INT(13*RN)+1
1230 ON M GOSUB 1960,1980,2000,2020,2040,2060,2080,2100,2140,2170,2200,2230,2260,2290,2320
1240 FOR Z=5 TO 25 STEP 5
1250 CALL GCHAR(12,Z,K)
1260 IF X<>KK THEN 1280
1270 GOTO 1300
```


TI-99/4A PROGRAM

```

1200 NEXT Z
1290 GOTO 1350
1300 FOR T=5 TO 25 STEP 5
1310 CALL GCHAR(10,T,K)
1320 IF K<P THEN 1340
1330 GOTO 1220
1340 NEXT T
1350 QW=ASC(STR$(Q/5))
1360 CALL HCHAR(7,Q,QW)
1370 IF QW<49 THEN 1420
1380 POINTS=10
1390 BET=0
1400 K=0
1410 GOTO 1650
1420 GOSUB 2690
1430 IF POINTS<1 THEN 1700
1440 A$="ENTER BET"
1450 RW=20
1460 F=4
1470 GOSUB 2470
1480 CALL HCHAR(20,15,32)
1490 CALL KEY(0,K,S)
1500 IF (S=0)+(K<49)+(K>57) THEN 1490
1510 BET=K-48
1520 CALL HCHAR(20,15,K)
1530 POINTS=POINTS-BET
1540 GOSUB 2690
1550 CALL HCHAR(20,4,32,6)
1560 CALL KEY(0,K,S)
1570 IF S=0 THEN 1560
1580 IF (K=69)+(K=88) THEN 1590 ELSE 1560
1590 CALL GCHAR(10,0-5,H)
1600 IF K<P THEN 1650
1610 GOSUB 2370
1620 GOSUB 1850
1630 GOSUB 2520
1640 GOTO 1700
1650 GOSUB 2370
1660 GOSUB 2750
1670 GOSUB 1850
1680 GOSUB 1810
1690 NEXT Q
1700 A$="AGAIN (Y/N)"
1710 RW=24
1720 F=3
1730 GOSUB 2470
1740 GOSUB 2690
1750 CALL KEY(0,K,S)
1760 IF S=0 THEN 1750
1770 GOSUB 3100
1780 IF K=89 THEN 1830
1790 IF K=78 THEN 2460
1800 GOTO 1750
1810 FOR I=0 TO 30 STEP 5
1820 CALL SOUND(-99,2000,I,-4,1)
1830 NEXT I
1840 RETURN
1850 CALL HCHAR(10,0,P)
1860 CALL HCHAR(10,0+1,PP)
1870 RETURN
1880 KK=144
1890 RETURN
1900 KK=148
1910 RETURN
1920 KK=152
1930 RETURN
1940 KK=156
1950 RETURN
1960 P=49
1970 PP=48
1980 RETURN
1990 P=50
2000 GOSUB 2350
2010 RETURN
2020 P=51
2030 GOSUB 2350
2040 RETURN
2050 P=52
2060 GOSUB 2350
2070 RETURN
2080 P=53
2090 GOSUB 2350
2100 RETURN
2110 P=54
2120 GOSUB 2350
2130 RETURN
2140 P=55
2150 GOSUB 2350
2160 RETURN
2170 P=56
2180 GOSUB 2350
2190 RETURN
2200 P=57
2210 GOSUB 2350
2220 RETURN
2230 P=74
2240 GOSUB 2350
2250 RETURN
2260 P=81
2270 GOSUB 2350
2280 RETURN
2290 P=75
2300 GOSUB 2350
2310 RETURN
2320 P=65
2330 GOSUB 2350
2340 RETURN
2350 PP=128
2360 RETURN
2370 CALL HCHAR(10,0,128)
2380 CALL HCHAR(10,0+1,128)
2390 CALL HCHAR(11,0,128)
2400 CALL HCHAR(11,0+1,128)
2410 CALL HCHAR(12,0,KK)
2420 CALL HCHAR(12,0+1,KK+1)
2430 CALL HCHAR(13,0,KK+2)
2440 CALL HCHAR(13,0+1,KK+3)
2450 RETURN
2460 GOTO 3000
2470 FOR I=1 TO LEN(A$)
2480 A=ASC(SEG$(A$,I,1))
2490 CALL HCHAR(RW,F+I,A)
2500 NEXT I
2510 RETURN
2520 A$="YOU GET NOTHING FOR A PAIR"
2530 POINTS=0

```

```

2540 FOR I=30 TO 0 STEP -3
2550 CALL SOUND(-99,111,1,-1,1,2220,1,4000,1)
2560 NEXT I
2570 FOR I=0 TO 30
2580 CALL SOUND(-99,1000,1,-7,1)
2590 NEXT I
2600 RW=6
2610 CALL HCHAR(7,1,32,25)
2620 F=3
2630 GOSUB 2470
2640 A$="NOT IN THIS GAME"
2650 RW=8
2660 F=3
2670 GOSUB 2470
2680 RETURN
2690 CALL HCHAR(20,24,32,5)
2700 FOR I=1 TO LEN(STR$(POINTS))
2710 PS=ASC(SEG$(STR$(POINTS),I,1))
2720 CALL HCHAR(20,24+1,PS)
2730 NEXT I
2740 RETURN
2750 IF K<49 THEN 2770
2760 H=56
2770 IF K<69 THEN 2880
2780 IF P=65 THEN 2980
2790 IF P<75 THEN 2810
2800 IF H=65 THEN 2990 ELSE 2980
2810 IF P<81 THEN 2830
2820 IF (H=65)+(H=75) THEN 2990 ELSE 2980
2830 IF P<74 THEN 2850
2840 IF (H=74)+(H=65) THEN 2990 ELSE 2980
2850 IF P<49 THEN 2870
2860 IF H=64 THEN 2990 ELSE 2980
2870 IF P>H THEN 2980 ELSE 2990
2880 IF P=65 THEN 2990
2890 IF P<75 THEN 2910
2900 IF H=65 THEN 2980 ELSE 2990
2910 IF P<81 THEN 2930
2920 IF (H=65)+(H=75) THEN 2980 ELSE 2990
2930 IF P<74 THEN 2950
2940 IF (H=74)+(H=65) THEN 2980 ELSE 2990
2950 IF P<49 THEN 2970
2960 IF H=64 THEN 2980 ELSE 2990
2970 IF P>H THEN 2990
2980 POINTS=POINTS+(2*BET)
2990 RETURN
3000 CALL CLEAR
3010 CALL SCREEN(2)
3020 PRINT "*****"
3030 PRINT "*****"
3040 PRINT "*****"
3050 PRINT "*****"
3060 A$="TOP SCORER "&B$
3070 RW=3
3080 F=6
3090 GOSUB 2470
3100 CL=INT(14*RND)+3
3110 CALL COLOR(12,CL,CL)
3120 FOR D=1 TO 50
3130 NEXT D
3140 FOR J=2 TO 8
3150 CALL COLOR(J,CL,1)
3160 NEXT J
3170 GOTO 3100
3180 IF POINTS<HPPOINTS THEN 3260
3190 HPPOINTS=POINTS
3200 IF Q<25 THEN 3220
3210 GOSUB 860
3220 IF HPPOINTS<46 THEN 3250
3230 X$="MAXIMUM "&B$
3240 GOTO 3270
3250 X$="HI SCORE "&B$
3260 POINTS=10
3270 RETURN

```

HIGH QUALITY TI-99/4A SOFTWARE PACKAGED IN AUDIO WALLET

ADVENTURE MANIA. (BASIC, NO COMMAND MODULE NEEDED) £5.95
H.C.W. GAVE THIS FULL GRAPHIC AND TEXT ADVENTURE 4 STARS.
CAN YOU DISCOVER THE ULTIMATE QUEST IN THIS SPECTACULAR MISSION. (30 LOCATIONS)

MANIA. (BASIC, NO COMMAND MODULE NEEDED) £5.95
H.C.W. GAVE 5 STARS AND SAID "I WOULD CERTAINLY BUY MANIA". FULLY GRAPHIC AND TEXT
ADVENTURE. CAN YOU SURVIVE IN MANIA AND ESCAPE TO THE CORRIDORS OF TIME...

LIONEL AND THE LADDERS (EXTENDED BASIC) £7.95
THE VERY BEST FULLY ANIMATED SPRITE GRAPHIC ADVENTURE YOU HAVE EVER SEEN ON
YOUR TI-99/4A. HELP OUR HERO LIONEL AS HE JOURNEYS THROUGH THE LABYRINTH IN
SEARCH OF HIS CAPTURED PRINCESS. BUT MAKE SURE HE DOES NOT GET CAUGHT BY THE SONS
OF MARS, THE SHADES OF WRATH OR THE DEATH STARS. LIONEL WILL CAPTIVATE YOU WITH
HIS STYLE. SEE OUR HALF PAGE ADVERTISEMENT IN LAST WEEK'S ISSUE OF HOME COMPUTING
WEEKLY.

★ AVAILABLE SHORTLY "BENEATH THE STARS" ★



Games to remember

from
**INTRIGUE
SOFTWARE**

SEND P.O./CHEQUE TO:
INTRIGUE SOFTWARE
CRANBROOK ROAD TENTERDEN KENT TN30 6UJ
WE WILL SEND YOUR GAMES BY RETURN, POST FREE.
DEALER ENQUIRIES WELCOME. TEL: 05806 4726

FLUFFY ANDROID SOFTWARE CONTACT

The Software Manager,
19 Ravenswood, Titchfield Common,
Fareham.

SPECTRUM PROGRAMMERS!
HAVE YOU WRITTEN A PROGRAM WITH
THAT LITTLE BIT EXTRA?

IF SO THEN WE WANT TO KNOW
ABOUT YOU!

TOP ROYALTIES WILL BE PAID
FOR FIRST CLASS GAMES
AND OTHER SOFTWARE!

SEND TAPE AND FULL DETAILS OF YOUR
PROGRAM TO ABOVE ADDRESS.

TI-99/4A

ARCADE HARDWARE

211 Horton Road, Fallowfield, Manchester, M14 7BE.
For Access orders and enquiries Tel. 061 225 2248

We have available for your TI99/4A The NEW TI games modules.
Moonwine, Hopper, MAXSKN, Slynoids, Space Bandit, Sewermania,
Bigfoot, Superfly, Jambreaker, Microsurgeon, Burgertime. All at
£27.50. From Atari : Donkey Kong (£28.95), Pac-man, Defender
(£18.95).

Arcaae Joystick Genuine steel joystick £22.00
Books

Introduction to Assembly Language for the TI Home Computer.
Priced at £16.95.

Programs for the TI Home Computer.
50 programs for the TI99/4A. Priced at £14.95
All prices inclusive of p&p and V.A.T.
Phone to check availability of ExBas & Mini Memory.

Charming TI-99/4A £3.20

SoftTI, 14 Station Rd, Brough,
N. Humberside

Ever fancied yourself as Prince Charming? Well, here's your chance. Wicked Count Bludnok has abducted Princess Softi and you must rescue her.

The game starts in the catacombs, a grid with numbered and lettered axes. You have a limited time to type in co-ordinates to open tombs and find matching pairs of items useful for the rescue. As each item only appears on the screen until another co-ordinate is entered or items matched, quick reactions and good memory are required.

How well you fare determines in which of three chambers you will find the princess. You may, however, wish to forget her and

make for the money — or perhaps both! Various guards and obstacles appear in your path and careful use of items from the catacombs is essential for survival.

Excellent use of the TI's capabilities is made, but during one of many attempts, 'Memory Full' brought the game to an abrupt end.

Feminists, grinding their teeth at such a male chauvinistic theme, will be pleased to learn that there's a version available in which the roles are reversed.

J.W.

instructions	95%
playability	90%
graphics	95%
value for money	100%



Spy's Demise TI-99/4A £8

Challenger, from Stainless Software, 10 Alstone Rd, Stockport, Cheshire SK4 5AH

This game makes a welcome change from others based wholly on manual dexterity. Its theme is typically American: you against the KGB. The documentation leads you into the Ukrainian diplomatic mission in Pyongyang. Ahem. While tipling in the Bangkok Hilton you have discovered that the KGB has the key to a fortune in computer data in the form of an encoded message. Your task is to avoid the spritely guards through the 11 floors of the diplomatic mission, not once but nine times, building up a score and piecing together the message as it is revealed to you.

If you are one of two UK solvers of the code you can win free software.

You use the joystick or the keyboard's s and d keys to control movement, while the guards patrol continuously. You can pause by pressing p or the fire button, but this is very difficult, needing quick mental reactions.

Graphics are quite good, and interest is held very well indeed. Needs Extended BASIC. P.B.

instructions	90%
playability	90%
graphics	80%
value for money	75%



Breakout/ Middle Kingdom Dragon £7.95

Dragon Data, Kenfig Industrial Estate, Margam, Port Talbot, West Glam SA13 2PE

Breakout is a machine code version of the arcade game. You must try to keep the ball in play while knocking out bricks from a descending wall. You can use a joystick or keys to control the bat. There are five skill levels and a high-score chart. A very enjoyable game which loaded with no problems.

The second game is also in machine code and is a graphic adventure game. At the start you choose the skill level and the speed at which messages are to be displayed on the screen.

The object is to return magic rings to the sanctuary to become the ruler of middle kingdom. You must search the catacombs, temple and pyramid before your strength runs out. Many evil beings live here and they must be defeated. Scattered objects will help you in your quest.

The main display is a bird's eye view of the part of the building you are in. Movement is controlled by the arrow keys.

This is a thoroughly enjoyable adventure game. Both games are very entertaining. J.E.M.

instructions	85%
playability	85%
graphics	90%
value for money	85%



Join the quest

Overcome obstacles to reach
your goal in this batch of
adventure games. Our panel of
experts have got there first

Solitaire/ Tower of Hanoi Sord M5 £8.95

Computer Games, CGL House, Goldings Hill, Loughton, Essex IG10 2RR

Two games at £8.95 for the standard Sord M5 with BASIC-I. I wasn't expecting much from these programs as they are written in 4K and when I loaded them I was right. They are very simple games.

Solitaire is the traditional board game where you try and clear as many pieces as you can by jumping over the ones immediately next to them.

Tower of Hanoi is also well known. You've probably seen it published for the more popular micros in magazines. It's the

game in which you move the pyramid-shaped edifice from one position to another, but only one tier can be moved at a time and you can only transfer smaller tiers on top of larger ones.

Both programs were well fool-proofed and I could not crash them. Considering the limitations of BASIC-I, they are quite good and you may find studying these programs more beneficial than reading the rather poorly written BASIC-I manual. In any case I must confess I did enjoy these games and at £8.95 for two, you can't be too critical can you? S.S.

instructions	95%
playability	60%
graphics	30%
value for money	60%



Tumbledown Tower TI-99/4A £5.55

Solway, 6 Curzon St, Maryport, Cumbria CA15 6LL

This is a fairly basic, text only adventure game in which you explore a derelict tower. Although the layout of the rooms and items in them remains the same in every run, it's still good fun to play.

You are given limited stamina to last each game: your stamina drains more rapidly as you collect various artefacts. A strange assortment of monsters is also provided to sap strength whilst wandering around the tower. You are always aggressive to them unless commanded otherwise. Consequently, amusing comments such as "you just killed a poor little bewildered dalek" appear on the screen.

Commands are input as a single letter which means the computer has a vocabulary limited to a few verbs and directions. You have the option to quit at any time, but your score is calculated before the final decision is taken.

The ultimate quest of the adventure is not described in the instructions, but takes little imagination to guess. Then it's just a matter of whether or not you have enough stamina to last the course. J.W.

instructions	80%
playability	65%
graphics	N/A
value for money	60%



Take over the board before your opponent

Use your skill and strategy to
cover the screen in your own
colour in this board game
from Dave Martin for the
unexpanded TI-99/4A

Hints on conversion

TI BASIC is fairly standard with two main exceptions. Single statement lines are used and subroutines for graphics and sound are available, prefixed with CALL, as detailed below.

CALL CLEAR clears screen

CALL SCREEN defines screen colour

CALL CHAR defines a character with a 16-digit hex string from an eight by eight grid. Other machines will use a different method

CALL COLOUR (character set, foreground, background) sets colour of character

CALL HCHAR (row, column, character code, number of repeats) places character on screen at co-ordinates specified and optionally repeats character horizontally

CALL VCHAR similar to HCHAR but repeats character vertically

CALL GCHAR (row, column, variable) similar to a screen PEEK on other machines. Gives the ASCII code at the given screen co-ordinates

CALL SOUND (D,F,V) gives sound of duration D, frequency F and volume V. Three sounds can be produced at once

This is a game of strategy for two players, the object being to win as many squares on an 8 x 8 grid as possible, the one gaining the most being the winner.

Each player starts with two squares in the centre of the grid. To play you have to trap opposition squares between the square you are adding to the grid and one of your existing squares, horizontally, vertically or diagonally (it is only a valid move if you do so). The trapped squares then become yours and change to your colour.

You play the game like this:

First select the column number (1-9), the column number flashes, then the row (A-H).

If the square is already occupied then nothing happens and it is still your turn.

If it is an invalid move then the selected square is displayed

How it works

100-320 define graphics

330-420 set up frame

430-520 number frame

530-780 set up start

790-1040 select square

1050-1510 check for trapped squares

1520 detects invalid move

1530-1560 changes over squares

1570-1630 changes over scoring variables

1640 detects end of game

1650-1680 changes over scoring variables

1690-1800 displays squares and scores

1810-1880 end of game sub-routine

1890-1990 removes invalid move

2000-2170 running score sub-routine

and then erased and it is still your turn.

Either of these conditions can be used to cancel an incorrect column choice.

If it is a valid move then the trapped squares are changed to your colour, the running score is updated and the 'turn' indicator changes to the opposition colour and it is his turn.

The game is finished when the grid is full. If a player cannot go then he must press 'N' which changes the colour of the 'turn' indicator and it is then his opponents turn.

Variables

A,B,AA used to change over squares

BAD used to detect invalid move

NG number of goes

R row number

C column number

RR old row number

CC old column number

NR & NC used in detecting trapped squares

SQUARES count of trapped squares

NOTE tone played when squares changed

NUM used in flashing column number subroutine

GAIN,LOSS,RED,WHITE used in scoring

GAINC used in changing over scoring to next player

RED1,RED2,WHITE1,WHITE2 used in running score display

```
120 CALL SCREEN(5)
140 CALL CLEAR
150 REM DEFINE FRAME*****
160 CALL CHAR(136,"FF80808080808080")
170 CALL CHAR(137,"FF01010101010101")
180 CALL CHAR(138,"80808080808080FF")
190 CALL CHAR(139,"01010101010101FF")
200 CALL COLOR(14,2,4)
210 REM DEFINE WHITES*****
220 CALL CHAR(128,"FF80808080808080")
230 CALL CHAR(129,"FF01010101010101")
240 CALL CHAR(130,"80808080808080FF")
250 CALL CHAR(131,"01010101010101FF")
260 CALL COLOR(13,2,16)
270 REM DEFINE REDS*****
280 CALL CHAR(144,"FF80808080808080")
290 CALL CHAR(145,"FF01010101010101")
300 CALL CHAR(146,"80808080808080FF")
310 CALL CHAR(147,"01010101010101FF")
320 CALL COLOR(15,2,7)
330 REM CALL FRAME*****
340 CALL CLEAR
350 FOR I=9 TO 23 STEP 2
360 FOR Z=6 TO 20 STEP 2
370 CALL HCHAR(Z,I,136)
380 CALL HCHAR(Z,I+1,137)
390 CALL HCHAR(Z+1,I,138)
400 CALL HCHAR(Z+1,I+1,139)
410 NEXT Z
420 NEXT I
430 REM NUMBER FRAME*****
440 FOR I=2 TO 8
450 CALL COLOR(I,16,1)
460 NEXT I
```

```
470 FOR I=9 TO 23 STEP 2
480 CALL HCHAR(5,I,(I+1)/2+44)
490 NEXT I
500 FOR Z=6 TO 20 STEP 2
510 CALL HCHAR(Z,8,Z/2+62)
520 NEXT Z
530 REM SET UP START*****
540 CALL HCHAR(12,3,144)
550 CALL HCHAR(12,27,128)
560 CALL HCHAR(12,5,48)
570 CALL HCHAR(12,6,50)
580 CALL HCHAR(12,29,48)
590 CALL HCHAR(12,30,50)
600 CALL HCHAR(12,15,128)
```

```
610 CALL HCHAR(12,16,129)
620 CALL HCHAR(13,15,130)
630 CALL HCHAR(13,16,131)
640 CALL HCHAR(14,17,128)
650 CALL HCHAR(14,18,129)
660 CALL HCHAR(15,17,130)
670 CALL HCHAR(15,18,131)
680 CALL HCHAR(12,17,144)
690 CALL HCHAR(12,18,145)
700 CALL HCHAR(13,17,146)
710 CALL HCHAR(13,18,147)
720 CALL HCHAR(14,15,144)
730 CALL HCHAR(14,16,145)
740 CALL HCHAR(15,15,146)
750 CALL HCHAR(15,16,147)
```


TI-99/4A PROGRAM

```

760 NG=0
765 TONE=440
770 GAIN=2
780 LOSS=2
790 REM SELECT POSITION*****
800 A=144
810 B=128
820 CALL HCHAR(24,14,84)
830 CALL HCHAR(24,15,79)
840 CALL HCHAR(24,17,71)
850 CALL HCHAR(24,18,79)
860 CALL KEY(0,K,S)
870 CALL HCHAR(24,12,A)
880 IF S=0 THEN 860
890 IF K=78 THEN 1540
900 NUMB=K
910 C=(K-48)*2+7
920 IF (C<9)+(C>23) THEN 860
930 CALL KEY(0,K,S)
940 CALL HCHAR(5,C,32)
950 CALL HCHAR(5,C,NUMB)
960 IF S=0 THEN 930
970 R=(K-64)*2+4
980 RR=R
990 CC=C
1000 IF (R<6)+(R>20) THEN 930
1010 CALL GCHAR(R,C,UU)
1020 IF (UU=144)+(UU=128) THEN 860
1030 GOSUB 1690
1040 LOSS=LOSS+1
1050 REM CHECK FOR LINES***
1060 SQUARES=0
1070 BAD=0
1080 FOR I=1 TO 8
1090 ON I GOTO 1100,1130,1160,1190,
1220,1250,1280,1310
1100 NR=-1
1110 NC=0
1120 GOTO 1330
1130 NR=-1
1140 NC=1
1150 GOTO 1330
1160 NR=0
1170 NC=1
1180 GOTO 1330
1190 NR=1
1200 NC=1
1210 GOTO 1330
1220 NR=1
1230 NC=0
1240 GOTO 1330
1250 NR=1
1260 NC=-1
1270 GOTO 1330
1280 NR=0
1290 NC=-1
1300 GOTO 1330
1310 NR=-1
1320 NC=-1
1330 CALL GCHAR(R+2*NR,C+2*NC,X)
1340 IF X=B THEN 1370
1350 IF X=A THEN 1410
1360 GOTO 1480
1370 SQUARES=SQUARES+1
1380 R=R+2*NR
1390 C=C+2*NC
1400 GOTO 1330
1410 IF SQUARES=0 THEN 1480
1420 BAD=1
1430 GOSUB 1690
1440 R=R-2*NR
1450 C=C-2*NC

```

```

1460 SQUARES=SQUARES-1
1470 GOTO 1410
1480 R=RR
1490 C=CC
1500 SQUARES=0
1510 NEXT I
1520 IF BAD=0 THEN 1890
1530 NG=NG+1
1540 AA=A
1550 A=B
1560 B=AA
1570 IF A=128 THEN 1580 ELSE 1610
1580 RED=GAIN
1590 WHITE=LOSS
1600 GOTO 1630
1610 RED=LOSS
1620 WHITE=GAIN
1630 GOSUB 2010
1640 IF NG=60 THEN 1820
1650 GAIN=GAIN
1660 GAIN=LOSS
1670 LOSS=GAIN
1680 GOTO 860
1690 REM CALL COLORS*****
1700 CALL HCHAR(R,C,A)
1710 CALL HCHAR(R,C+1,A+1)
1720 CALL HCHAR(R+1,C,A+2)
1730 CALL HCHAR(R+1,C+1,A+3)
1740 CALL SOUND(-300,TONE,0)
1750 TONE=TONE+110
1760 IF TONE>2200 THEN 1770 ELSE 1780
1770 TONE=440
1780 GAIN=GAIN+1
1790 LOSS=LOSS-1
1800 RETURN
1810 REM END ROUTINE*****
1820 PRINT "AGAIN? (Y/N) "
1830 CALL KEY(0,K,S)
1840 IF S=0 THEN 1830
1850 IF K=89 THEN 330
1860 IF K=78 THEN 1880
1870 GOTO 1830
1880 END
1890 REM REMOVE BAD MOVE***
1900 R=RR
1910 C=CC
1920 NW=0
1930 CALL HCHAR(R,C,136)
1940 CALL HCHAR(R,C+1,137)
1950 CALL HCHAR(R+1,C,138)
1960 CALL HCHAR(R+1,C+1,139)
1970 CALL SOUND(100,-2,0,220,15,880,25)
1980 GAIN=GAIN-1
1990 GOTO 860
2000 REM RUNNING SCORE*****
2010 IF RED<10 THEN 2070
2020 RED1=VAL(SEG$(STR$(RED),1,1))+48
2030 RED2=VAL(SEG$(STR$(RED),2,1))+48
2040 CALL HCHAR(12,5,RED1)
2050 CALL HCHAR(12,6,RED2)
2060 GOTO 2090
2070 CALL HCHAR(12,6,RED+48)
2080 CALL HCHAR(12,5,48)
2090 IF WHITE<10 THEN 2150
2100 WHITE1=VAL(SEG$(STR$(WHITE),1,1))+48
2110 WHITE2=VAL(SEG$(STR$(WHITE),2,1))+48
2120 CALL HCHAR(12,29,WHITE1)
2130 CALL HCHAR(12,30,WHITE2)
2140 GOTO 2170
2150 CALL HCHAR(12,30,WHITE+48)
2160 CALL HCHAR(12,29,48)
2170 RETURN

```


TI-99/4A PROGRAM

```

10 CALL CLEAR
20 CALL SCREEN(2)
30 FOR I=1 TO 8
40 CALL COLOR(1,16,1)
50 NEXT I
60 PRINT " PEARL DIVERS BY PHIL ORD."
70 PRINT " INSTRUCTIONS" USE DIRECTION KEYS(ESDX) "TO GUIDE YOUR D
IVER AROUND"
80 PRINT " THE SEA,PICKING UP PEARLS" BEWARE OF THE RANDOMLY PLACED
MINES AND THE
90 PRINT " MAN-EATING SHARKS,IF YOU" MAKE CONTACT WITH THESE YOUR D
IVER,IS LOST"
100 PRINT " PRESS ENTER TO CONTINUE"
110 CALL KEY(0,K,S)
120 IF S=0 THEN 110
130 IF K=30 THEN 140
140 CALL CLEAR
150 PRINT " IF YOU LOSE 5 DIVERS" THE GAME ENDS AND YOU LOSE" ALL THE PE
ARLS YOU COLLECTED"
160 PRINT "YOU LOSE PEARLS FOR STAYING" INTO THE BARRACUDAS OR CORAL"
MEN YOU COLLECT 400"
170 PRINT "PEARLS,THE BLACK CAVE HIDING" THE TREASURE WILL OPEN,"
180 PRINT " WARNING!! IF YOU TOUCH THE" BLACK WALLS OF THE CAVE," TH
E GAME WILL END"
190 PRINT " PRESS ENTER TO CONTINUE"
200 CALL KEY(0,K,S)
210 IF S=0 THEN 200
220 IF K=30 THEN 230
230 CALL CLEAR
240 CALL SCREEN(2)
250 MEN=5
260 P=0
270 PP=0
280 REM DEFINE CHARACTERS
290 CALL CHAR(40,"0000001010000000")
300 CALL CHAR(41,"11925430FF305492")
310 CALL CHAR(42,"FFFFFFFFFFFFFF")
320 CALL CHAR(43,"0042001010004200")
330 CALL CHAR(44,"FFFFFFFFFFFFFF")
340 CALL CHAR(104,"02077F0FF1F7C00")
350 CALL CHAR(105,"0102FC0F0FC0201")
360 CALL CHAR(106,"000130FC30010000")
370 CALL CHAR(112,"FF7F3F1F0F070301")
380 CALL CHAR(113,"FFFFFFFFFFFFFF")
390 CALL CHAR(114,"FFFFFFFFFFFFFF")
400 CALL CHAR(115,"FFFF000000000000")
410 CALL CHAR(120,"1010305492204444")
420 CALL CHAR(121,"10C0241F24C01000")
430 CALL CHAR(122,"222214492A1C0000")
440 CALL CHAR(123,"00001324F0241300")
450 CALL CHAR(129,"FFFFFFFFFFFFFF")
460 CALL CHAR(136,"0000001010305492")
470 CALL CHAR(144,"007E7E1010101000")
480 CALL CHAR(152,"FFFFFFFFFFFFFF")

```

Using the direction keys, E,S,D, and X, collect as many pearls as you can without being eaten by the sharks or blown up by mines. There is also a treasure hidden in a secret cave

Variables

R - C row and column of diver
R2 - C2 row and column after diver
DIV diver
MEN number of divers
P pearls
PP penalty points
TR treasure

```

490 REM DEFINE COLOURS
500 CALL COLOR(2,15,6)
510 CALL COLOR(9,7,1)
520 CALL COLOR(10,2,6)
530 CALL COLOR(11,13,15)
540 CALL COLOR(12,2,6)
550 CALL COLOR(13,6,6)
560 CALL COLOR(14,2,15)
570 CALL COLOR(15,2,11)
580 CALL COLOR(16,2,2)
590 REM SET UP SCREEN
600 CALL HCHAR(5,1,40,640)
610 CALL HCHAR(1,1,42,120)
620 RESTORE 710
630 FOR I=1 TO 24
640 READ A,B,C
650 CALL HCHAR(A,B,96,C)
660 NEXT I
670 FOR I=1 TO 3
680 READ A,B,C
690 CALL VCHAR(A,B,96,C)
700 NEXT I
710 DATA 1,1,3,2,1,3,3,1,3,4,1,6,5,1,6,6,1,6,7,1,6,8,1,6,9,1,6,10,1,6
720 DATA 11,1,6,12,1,6,13,1,6,14,1,5,15,1,2,16,1,2
730 DATA 17,1,2,18,1,11,19,1,2
740 DATA 20,1,3,21,1,6,22,1,11,23,1,2,24,1,2
750 DATA 4,31,21,4,32,21,7,30,12

```

which is revealed during the game.

How it works

10-220 instructions
230-270 set initial values
280-480 define characters
590-1200 set up screen graphics
1210-1450 move diver
1460-1500 diver at edge of screen or on surface of water
1510-2050 position of diver and consequences
2060-2830 end routines and score
2140-2300 reveals treasure cave

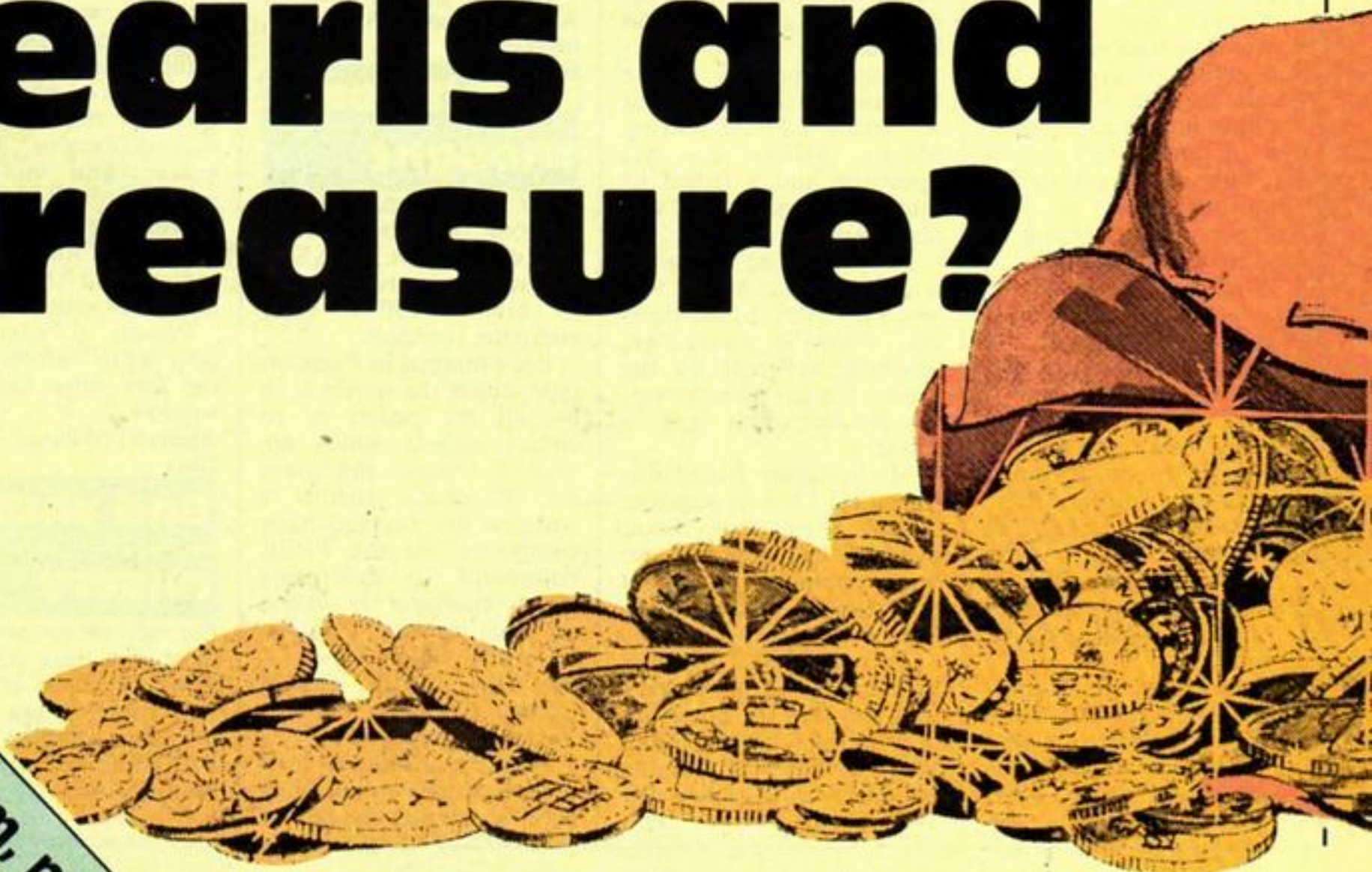
```

760 T=2
770 T=T+4
780 IF T>23 THEN 890
790 M=0
800 M=M+1
810 IF M=5 THEN 880
820 RANDOMIZE
830 U=INT(RND*21)+7
840 A=INT(RND*20)+8
850 CALL HCHAR(T,U,43)
860 CALL HCHAR(T+2,A,105)
870 GOTO 880
880 GOTO 770
890 CALL HCHAR(21,25,105,3)
900 CALL HCHAR(22,24,105,4)
910 CALL HCHAR(23,25,105,3)
920 CALL HCHAR(24,25,105,3)
930 CALL HCHAR(22,29,105)
940 CALL HCHAR(24,29,105)
950 RESTORE 1040
960 FOR I=1 TO 12
970 READ A,B,C
980 CALL HCHAR(A,B,104,C)
990 NEXT I
1000 FOR I=1 TO 12
1010 READ A,B,C
1020 CALL HCHAR(A,B,105,C)

```

Will you risk sharks and mines for pearls and treasure?

Type in Phil Ord's program, put on your diving gear and you're ready to face twin dangers for rich rewards



TI-99/4A PROGRAM

```

1030 NEXT I
1040 DATA 11,15,1,13,24,1,15,7,1,21,13,1,19,29,1,22,10,1
1050 DATA 5,16,1,8,22,1,9,11,1,16,19,1,17,26,1,19,21,1
1060 DATA 11,16,1,13,25,1,15,8,1,21,14,1,19,30,1,22,19,1
1070 DATA 5,17,1,8,23,1,9,12,1,16,20,1,17,27,1,19,22,1
1080 CALL HCHAR(4,25,112)
1090 CALL HCHAR(4,26,113,4)
1100 CALL HCHAR(3,27,114)
1110 CALL HCHAR(3,28,115)
1120 CALL HCHAR(3,26,136)
1130 CALL HCHAR(5,29,120)
1140 CALL KEY(0,K,S)
1150 IF S=0 THEN 1140
1160 R=5
1170 C=29
1180 X=1
1190 Y=0
1200 DIV=120
1210 REM MOVE DIVER

```

```

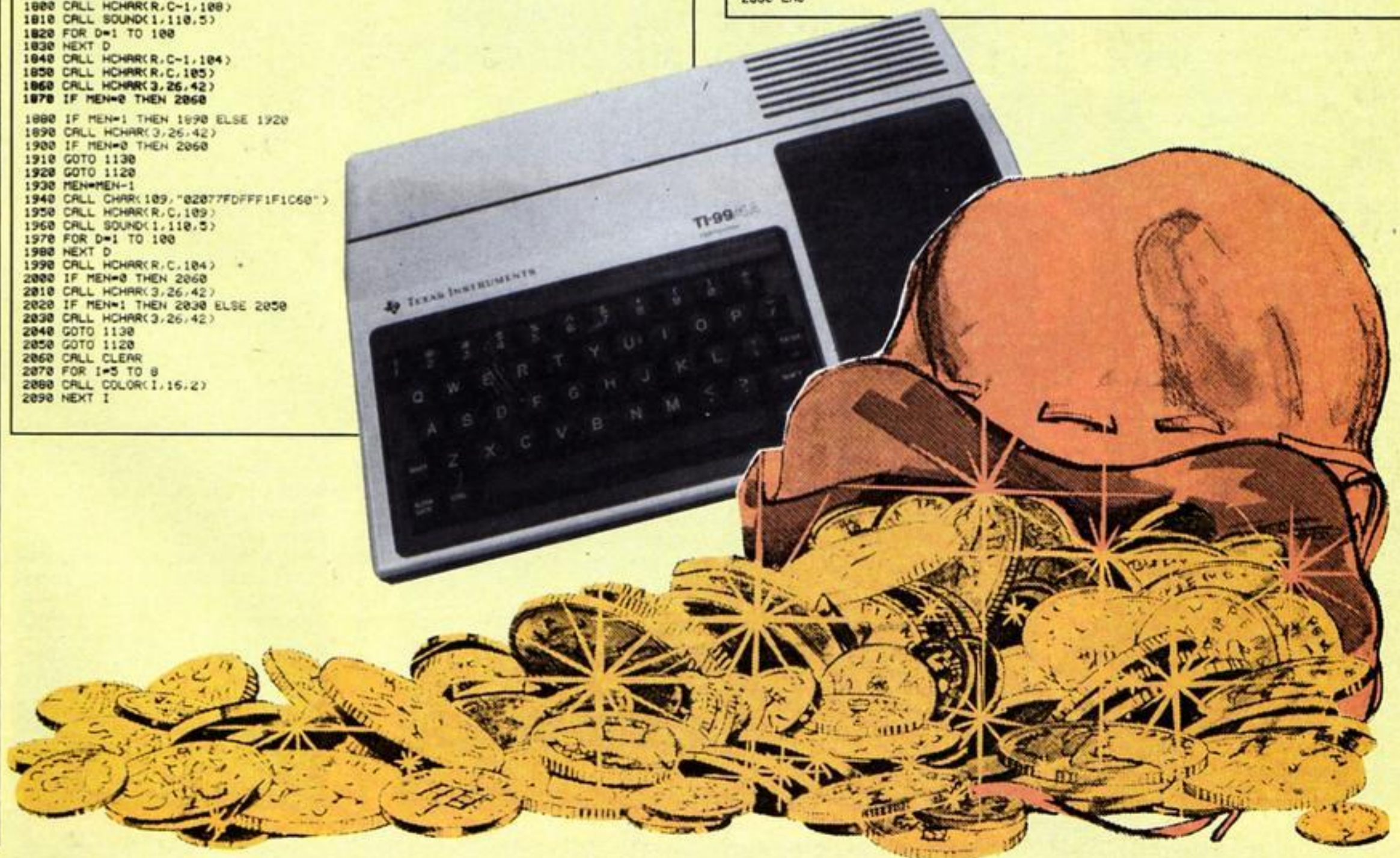
1220 CALL KEY(0,K,S)
1230 R2=R
1240 C2=C
1250 IF S=0 THEN 1440
1260 CALL SOUND(10,44733,0)
1270 IF K<>69 THEN 1310
1280 Y=-1
1290 X=0
1300 DIV=120
1310 IF K<>69 THEN 1350
1320 Y=1
1330 X=0
1340 DIV=122
1350 IF K<>69 THEN 1390
1360 Y=0
1370 X=-1
1380 DIV=123
1390 IF K<>69 THEN 1440
1400 Y=0
1410 X=1
1420 DIV=121
1430 REM MOVE DIVER
1440 R=R+Y
1450 C=C+X
1460 REM DIVER AT EDGE OF SCREEN
1470 IF R<5 THEN 1660
1480 IF R>24 THEN 1660
1490 IF C<1 THEN 1700
1500 IF C>32 THEN 1720
1510 REM CONTACTS OF DIVER AND CONSEQUENCES
1520 CALL GCHAR(R,C,PSN)
1530 CALL HCHAR(R2,C2,120)
1540 CALL HCHAR(R,C,DIV)
1550 IF PSN=105 THEN 1760
1560 IF PSN=104 THEN 1930
1570 IF PSN=40 THEN 1640
1580 IF PSN=96 THEN 2310
1590 IF PSN=126 THEN 1220
1600 IF PSN=106 THEN 2340
1610 IF PSN=144 THEN 2370
1620 IF PSN=43 THEN 2590
1630 IF PSN=152 THEN 2710
1640 P=P+1
1650 IF P=400 THEN 2140 ELSE 1220
1660 R=R+1
1670 GOTO 1730
1680 R=R-1
1690 GOTO 1730
1700 C=C+1
1710 GOTO 1730
1720 C=C-1
1730 CALL KEY(0,K,S)
1740 IF S=0 THEN 1730
1750 GOTO 1220
1760 MEN=MEN-1
1770 CALL CHAR(107,"40E0FEFFBFFB1E00")
1780 CALL CHAR(108,"80403F1F1F3F4000")
1790 CALL HCHAR(R,C,107)
1800 CALL HCHAR(R,C-1,108)
1810 CALL SOUND(1,110,5)
1820 FOR D=1 TO 100
1830 NEXT D
1840 CALL HCHAR(R,C-1,104)
1850 CALL HCHAR(R,C,105)
1860 CALL HCHAR(3,26,42)
1870 IF MEN=0 THEN 2060
1880 IF MEN=1 THEN 1890 ELSE 1920
1890 CALL HCHAR(3,26,42)
1900 IF MEN=0 THEN 2060
1910 GOTO 1130
1920 GOTO 1120
1930 MEN=MEN-1
1940 CALL CHAR(109,"02077FDFFF1F1C00")
1950 CALL HCHAR(R,C,109)
1960 CALL SOUND(1,110,5)
1970 FOR D=1 TO 100
1980 NEXT D
1990 CALL HCHAR(R,C,104)
2000 IF MEN=0 THEN 2060
2010 CALL HCHAR(3,26,42)
2020 IF MEN=1 THEN 2030 ELSE 2050
2030 CALL HCHAR(3,26,42)
2040 GOTO 1130
2050 GOTO 1120
2060 CALL CLEAR
2070 FOR I=5 TO 8
2080 CALL COLOR(1,16,2)
2090 NEXT I

```

```

2100 PRINT " YOU RAN OUT OF DIVERS" AND LOST ALL YOUR PEARLS" PLAY
AGAIN Y OR N"
2110 CALL KEY(0,K,S)
2120 IF S=0 THEN 2110
2130 IF (K=89)+(K=121) THEN 230 ELSE 2060
2140 CALL HCHAR(12,3,120,4)
2150 CALL HCHAR(11,3,120)
2160 CALL HCHAR(10,3,120,3)
2170 CALL HCHAR(9,5,120)
2180 CALL HCHAR(8,4,120,2)
2190 CALL COLOR(15,2,11)
2200 CALL HCHAR(7,4,144)
2210 CALL HCHAR(6,3,152,3)
2220 CALL HCHAR(7,3,152)
2230 CALL HCHAR(7,5,152)
2240 CALL HCHAR(6,3,152)
2250 CALL HCHAR(9,2,152,3)
2260 CALL HCHAR(11,4,152,3)
2270 CALL VCHAR(7,6,152,4)
2280 CALL VCHAR(10,2,152,3)
2290 CALL HCHAR(13,2,152,5)
2300 GOTO 1220
2310 PP=PP+100
2320 CALL SOUND(10,4000,3)
2330 GOTO 1220
2340 PP=PP+1000
2350 CALL SOUND(10,2000,3)
2360 GOTO 1220
2370 TR=100000
2380 T=TR+P*100-PP
2390 FOR I=660 TO 1100 STEP 110
2400 FOR J=110 TO 660 STEP 110
2410 CALL SOUND(10,1,5)
2420 CALL SOUND(10,J,5)
2430 NEXT J
2440 NEXT I
2450 CALL CLEAR
2460 CALL SCREEN(15)
2470 FOR I=1 TO 16
2480 CALL COLOR(1,16,2)
2490 NEXT I
2500 PRINT " WELL DONE YOU FOUND THE"
2510 PRINT " TREASURE. WORTH $100,000"
2520 PRINT " YOU COLLECTED" IF "PEARLS" WORTH 100 DOLLARS EACH"
2530 PRINT " LESS" PP "DOLLARS" FOR LOST PEARLS"
2540 PRINT " FOR A TOTAL FORTUNE OF"
2550 PRINT "PLAY AGAIN Y OR N"
2560 CALL KEY(0,K,S)
2570 IF S=0 THEN 2560
2580 IF K=89 THEN 230 ELSE 2060
2590 MEN=MEN-1
2600 CALL HCHAR(R,C,41)
2610 FOR I=0 TO 30
2620 CALL SOUND(100,-7,1)
2630 NEXT I
2640 CALL HCHAR(R,C,120)
2650 IF MEN=0 THEN 2060
2660 CALL HCHAR(3,26,42)
2670 IF MEN=1 THEN 2680 ELSE 2700
2680 CALL HCHAR(3,26,42)
2690 GOTO 1130
2700 GOTO 1120
2710 FOR I=1 TO 16
2720 CALL SOUND(-100,-7,1)
2730 CALL COLOR(1,7,1)
2740 CALL COLOR(1,16,1)
2750 CALL COLOR(1,7,1)
2760 NEXT I
2770 FOR I=1 TO 16
2780 CALL COLOR(1,16,2)
2790 NEXT I
2800 CALL CLEAR
2810 PRINT " YOU CRASHED INTO THE" BLACK CAVE, YOU FORFEIT" ANY PE
ARLS YOU MAY"
2820 PRINT " HAVE COLLECTED" TRY AGAIN Y OR N,
2830 CALL KEY(0,K,S)
2840 IF S=0 THEN 2830
2850 IF (K=89)+(K=121) THEN 230 ELSE 2060
2860 END

```



Mission XK1 Dragon 32 £6.95

J. Morrison (Micros), 2 Glensdale St, Leeds LS9 9JJ

The first thing you notice about this game is the starting music which, although well done, sounds two octaves too low.

It is a version of the space rescue game and you must leave the mothership, dodge the meteor shower and land on the lunar surface. Unfortunately, there are no men to rescue in this version and the purpose of your mission is to collect fuel from underneath the planet's surface instead. Then on the return journey you have the option to dodge or shoot at the meteor shower before you dock with the

mothership again.

That is really all there is to this game and I feel it lacks the enchantment that makes some games compulsive. You have the option of joystick or keyboard control and whilst both are responsive I found the joystick to be easier.

The difficult part of the game is the docking, which is made so by the small area you have to aim for. Far from making the game more exciting I found this detracted from the game's appeal.

D.C.

instructions	65%
playability	60%
graphics	80%
value for money	65%



Cuber Dragon 32 £8

Tom Mix, 41 Truro Rd, St Austell, Cornwall PL25 5JE

A machine code arcade game in which you must help Cuber hop on to each of 28 cubes forming a pyramid to change them into the required colour.

Coloured balls bounce down the screen and the effect they have depends on their colour. If a blue ball reaches the bottom it turns into a viper which moves at random. Landing on this destroys one of your three lives. There is also a nasty piece of work called a Dork which descends at random and changes the cubes back to their original colour. Landing on this creature

gives you a handy bonus score. There are discs alongside the pyramid onto which you can jump and these are safe zones.

Three levels of play; at the start you are shown graphically what to avoid and what is beneficial. There is a short display of Cuber hopping around a mini pyramid.

The instructions on the insert explain the game comprehensively. Screen display is good. Loading was slow but caused no problems. An entertaining and amusing game.

J.E.M.

instructions	80%
playability	75%
graphics	80%
value for money	75%



Crash-Up Car Racing TI-99/4A £7

DEJ, from Stainless, 10 Alstone Rd, Stockport, Cheshire SK4 5AH

TI BASIC rarely lends itself to car-race games because of its slow execution speed and poor response to keyboard or joystick input. This program makes a valiant attempt to imitate arcade quality graphics but alas, fails miserably.

You can race against another player or the computer. The graphics display takes quite a time to set up, and has to be redrawn after every completed game. You can select 1 to 99 laps, accelerate and decelerate and speed and fuel are indicated on screen, albeit rather unclearly.

There are four reserve cars at your disposal; you will need them as it is easy to crash on the bends. There is a pit area in which you can stop, although if driving against the computer this will reduce your chance of winning to zero. You can even specify the efficiency of your pit crew.

The documentation makes this program appear more special than it is, and unfortunately this contributes to the overall impression that this is definitely not value for money.

P.B.

instructions	90%
playability	50%
graphics	40%
value for money	20%



There's more where these came from...

Yes, here are more programs
for your tried and tested old
faithfuls. They've all been
checked over by our regular
reviewers

Tukom's Kingdom TI-99/4A £6

Stainless, 10 Alstone Road, Stockport, Cheshire SK4 5AH

In this arcade/adventure game you take the rôle of a knight from Nige on a quest for the Great Stone, stolen by the Zok Monsters.

The first screen is a maze, with a Jump Spell, door, key and monster. You must collect the key before going through the door at the bottom of the screen. If you get the Jump Spell (by landing on a capital S near the bottom of the maze) you can jump to a random place in the

maze. This action can be carried out only once each time the spell is gained.

Unfortunately there is a bug: the monster sometimes moves over the spell, clearing it from the screen.

In screen two, you enter the dark forest. The only difference from the maze is that the forest has trees, not walls, randomly placed around the screen.

There are many other screens; throughout, your weapon of defence is your sword.

J.J.

instructions	70%
playability	80%
graphics	80%
value for money	80%



Bombard Sord M5 £5.95

CGL, CGL Hse, Goldings Hill, Loughton, Essex IG10 2RR

You are the pilot of a plane which is running out of fuel. Your height is decreasing with each pass across the screen. To land safely you must flatten the buildings by bombing them.

You've probably realised this is an M5 version of that classic arcade game. Although the program is written in BASIC-I, the sound is satisfyingly different, the plane looks like a plane and it rumbles across the screen as it should.

The action is relatively fast and I found the game fun, at least for the first few goes. Nice features such as a neat title page, instructions and various skill levels show that quite a bit of thought has gone into the design but ultimately it must be said that this is the type of game published in magazines within two weeks of a new machine's launch. It would never be sold commercially for the more popular machines.

It is sad this isn't the case with the M5 and you must fork out £5.95 for a rather simple, pure BASIC game instead of just typing it in.

S.S.

instructions	50%
playability	50%
graphics	70%
value for money	50%



Spy's Demise TI-99/4A + MiniMemory £8

Challenge, from Stainless, 10 Alstone Rd, Stockport, Cheshire SK4 5AH

This is the machine code version of the Extended BASIC game, and oddly enough easier to play. You are still at the Ukrainian diplomatic mission in Pyongyang, and you must cross 11 floors nine times, avoiding patrolling guards while building up a coded message — the solution of which could win you some software — and you still use either keyboard or joystick.

This time, however, the guards bounce off the top and bottom floors instead of wrapping round, which makes life more difficult, but your movement

control is far tighter, and when you use the option to pause everything stops, unlike the Extended BASIC version. It is this, more than anything, which makes the game easier and because there is less frustration, the enjoyment is greater.

Although there are really two games here — the on-screen battle with the guards and the code-breaking — the code could easily be obtained in both versions. However, the game would hold your attention and interest in its own right.

The graphic display is very similar to the Extended BASIC version, and shows better than average control of sprites. **P.B.**

instructions	90%
playability	100%
graphics	90%
value for money	85%



Lander/ Squash/ Raiders Sord M5 £12.95

CGL, CGL Hse, Golding Hill, Loughton, Essex IG10 2RR

This package consists of three games.

Lander is, as you can probably guess, the classic moon landing game. It's badly implemented and worst of all I found it impossible to land.

In Squash you move a bat up and down to keep a ball in court for as long as possible. Every time the ball hits the wall, your score is incremented. No knocking down bricks. Graphics and sound leaves a lot to be desired and the game itself is unoriginal and boring.

In Raiders, you control a

fighter plane moving left and right and shooting down the enemy planes which fly down screen. Although simple in concept, I found this game surprisingly addictive.

Since these are all written in BASIC-I to run in 4K, the graphics and sounds are very crude compared with the cartridge games available. What's more, they are all very unoriginal and, with the exception of the last one, very boring. Although together they only cost half as much as a cartridge, they are of very low quality and can not be recommended very highly. **S.S.**

instructions	60%
playability	40%
graphics	40%
value for money	40%



Electron Dragon 32 £8

Tom Mix Software, 41 Truro Rd, St Austell, Cornwall PL25 5JE

This package consists of four subgames for one or two players which must be successfully completed before you can move to the next level.

Each player starts with four lives. Between subgames a selection board is displayed for choice of game. Apparently you can use the right or left joystick, although I found I could use the right one.

The four games are: Prachnid, in which you must reach your transporter in the centre of the screen before you are shot or before the time runs out.

Beam Buggy in which your buggy and the computers leave vapour trails. The first buggy to crash into these loses.

Maze in which your tank must try to hit the computers three times before they destroy you.

Force Field in which you are trapped in a cylinder with a collapsing roof. You must shoot your way through before you are crushed to death.

Scores vary for each game. Extra lives are awarded if you reach certain scores.

Not very impressive graphics or games. Loading seemed to take a long time. **J.E.M.**

instructions	60%
playability	30%
graphics	40%
value for money	25%



Games, games and more games!

Here are five action-packed
arcade games which have
been thoroughly scrutinized
by our expert reviewers. Read
before you buy

Pengon Dragon 32 £8

Microdeal, 41 Truro Rd, St Austell, Cornwall PL25 5JE

This game for one player has Pengon the penguin travelling through the icy mazes trying to avoid or crush the nasty sea beasts.

There are five skill levels which govern speed of play. You can choose a black, buff or green background.

You must guide the penguin through the ice cube mazes. By using the fire button you can crush the cubes or push them at the nasties to sandwich them between two cubes. I never knew that penguins could be so vicious, but it is a case of you or them.

There are three easily identifiable magic cubes which cannot be crushed but can be pushed together for a bonus of 1,000 points and progression to the next screen.

At 10,000 points you get an extra penguin. The graphics are acceptable but I was disappointed to find that Pengon does not look much like a penguin. The program loaded with no problems but it did take a long time. Despite this I enjoyed helping Pengon in his travels and in his fight against the beasts. **J.E.M.**

instructions	75%
playability	75%
graphics	75%
value for money	70%



Galactic Encounter/ Checkers TI-99/4A £6

Stainless, 10 Alstone Rd, Stockport, Cheshire SK4 5AH

Galactic Encounter is a modern version of Othello for one or two players. In the one-player version, the computer takes the rôle of Evil Astex 99, who is fighting you in Sub Quadrant 4A. In the two-player version, player two takes the place of the computer.

The object of the game is to capture the opponent's troops. This is done by moving your troops into galaxies adjacent to those occupied by the opponent. The game finishes when there are no unoccupied galaxies, or if either player has no troops left, or if both players are unable to move.

A clever graphics effect is used to make the troops' eyes and feet move. I assume this is done by constantly changing their character codes.

The second game on this cassette, Checkers, is a variation of the old strategy game of Solitaire, except that the layout used is an 8 x 8 grid, with the centremost sixteen squares left blank.

Stainless is awarding a free program to the person who sends in a solution for Checkers before the end of the year. The prize will go to the best solution.

Good value. **J.J.**

instructions	75%
playability	90%
graphics	90%
value for money	90%



Progpak TI-99/4A £

T. Wilmott, 3 Somerset Place,
Somerset Bridge, Bridgewater
TA6 6LW

Games to suit everyone's taste. Ten in all, though I could only get nine to load. Nevertheless, quite remarkable value for a fiver. Here's a quick run down of the titles:

Stop Thief — a collect and chase game where a foreman wanders round scaffolding dropping fivers. Lucky you must pick them up, for as long as you can!

Base Attack — you have five laser cannon to defend against an alien attack. Very addictive.

Bat Blaster — self explanatory.

Hostile UFO's — aliens again, that swoop and bomb this time.

Treasure Seeker — dig around

searching for maps and treasure. **Cave Adventure** — the one that wouldn't load, but, from reading the instructions, would appear to be a standard adventure.

Alien Overrun — differently coloured aliens move across screen. Zap them to survive.

Robo-killer — takes some thinking about. Your task is to kill 12 robots by laying pillars and bombs, but they close in on you after every move.

Mystery Game — infuriating but entertaining.

Well there they are. What more could any software hungry TI owner want, excellent games at a giveaway price. **J.W.**

instructions	95%
playability	90%
graphics	80%
value for money	100%



Ophyss TI-99/4A £6

Stainless, 10 Alstone Rd, Stockport, Cheshire SK4 5AH

The object of this rather unusual game is to keep a number of newly-hatched ophyss eggs alive by steering them around a serpentarium. An ophyss is a snake-like creature found in the Antarctic islands. The young ophysses (or ophi?) must have nourishment, so they eat frogs.

In the two-player version a lifeline is displayed at each side of the screen. The length of this line shows how much nourishment each ophyss has, constantly decreasing. If an ophyss eats a frog, its lifeline returns to the maximum length and the ophyss grows by one unit. When it is 15 units long it expires, and a small

ophyss emerges from its den. This ophyss must also grow to 15 units.

The opponent's tail may also be eaten. This does not give any nourishment, but 20 points are gained. When an ophyss has been bitten, it is paralysed for one move, so if possible, it should be bitten again.

In the one-player version of the game, the computer controls the opponent's ophyss, and there are seven levels of difficulty.

Overall, a good game that would be even better if in machine code on another computer. **J.J.**

instructions	95%
playability	80%
graphics	90%
value for money	90%



Motor Cross/ Rally Cross TI-99/4A £7

Stainless, 10 Alstone Rd, Stockport, Cheshire SK4 5AH

These two programs are sold together for either joystick or keyboard use. Although good use has been made of the TI's BASIC facilities in an attempt to get the utmost speed, both programs are essentially identical. They differ mainly in the graphics used for gates, making this very expensive duplication.

Both are moderately challenging: you must drive around the screen, negotiating cones and trees and attempting to pass through eight gates. In Motor Cross only, the gates are numbered and must be negotiated sequentially. You have control of left and right turns only, which can be a little confusing as they are dependent upon the direction in which the car is heading.

There are five levels of play and your efforts are timed within the restrictions of TI BASIC. If you complete the course unscathed, you will receive a rating. I didn't even rate pathetic, as with the computer in control of the speed and no brakes I frequently became entangled with the trees.

If the games had differed more I would have considered this fair value for money.) **P.B.**

instructions	90%
playability	85%
graphics	75%
value for money	20%



Texas: the state of the stars

Here's a treat for owners of
the TI-99/4A: five new
programs reviewed by our
team

Rivets/ Charset 2 TI-99/4A £4.95

You need a good head for heights in Rivets, though whether you'll find it a truly riveting experience is another matter.

The game starts on the 21st floor of a partially constructed tower block. The site foreman has it in for you! When all your workmates have gone home he makes you collect up the rivets they have left lying around. Trouble is, there are banana skins scattered. Slipping on one sends you down onto the next floor and loses a life. The same happens if you miss your step on girders. When all your lives have

gone the building is displayed and you watch your death leap.

All this proceeds in a desperate race against time. There are five skill levels. As the game becomes more difficult you are given less time to finish and more banana skins to avoid.

Although fun at first, the game lacks sufficient variety to give it a lasting addictive quality.

Charset is a short program in which characters have been re-defined and their hexadecimal codes placed in DATA statements.) **J.W.**

instructions	85%
playability	40%
graphics	40%
value for money	50%



Escape from Mica TI-99/4A £4.95

PiKaDee Software, 35 Parker St, Preston, Lancs PR2 2AH

It's hard and dangerous work being a captive of the evil Count of Mica. To obtain your release all the treasures from each of three fields must be collected. Each is surrounded by a high wall and there are two blocks of wall inside the field.

The treasure is guarded by a hungry man-eating monster. Here the two inner walls prove useful because if you can lure the monster behind one he will momentarily stop. Then you take what you can before he becomes active again.

Four hyperspace pills have been provided to use in each field. Taking one will immediately transport you to another random location within the field. This proves very handy when the monster comes too close for comfort. Sometimes, however, you can find yourself re-appearing just as near, or closer!

There are four skill levels. At the easiest you score 15 points per item, but there are more items and large inner walls. At the most difficult you score 60 points per item, there's fewer items and much smaller walls. **J.W.**

instructions	90%
playability	60%
graphics	50%
value for money	65%



Now that summer's finally here, all the insects are out and about and making the most of the warmer weather. So are you in this game.

You are a caterpillar and you spend most of the lazy hazy days of summer eating leaves. Your particularly favourite patch is the mushroom patch.

Eating leaves makes you grow at a miraculous rate. You can almost see yourself expand — it's a good thing that doesn't happen in real life! You must avoid bumping into your own body or colliding with mushrooms, or you'll lose one of your three lives. Once all your lives have been used up, the game ends.

Another obstacle to beware is the hedge bordering the

How it works

10 go ahead to print instructions
15-60 first set of variables set-up
70 clear screen
80 set screen colour to black
90-110 set text to white
120-150 define colours
160-240 define characters
250-320 second set of variables set-up
330-360 hedge displayed
370-400 score printed
410-440 patch number printed
450-480 number of lives printed
490-610 mushrooms displayed
620-710 leaves displayed
720-730 display caterpillar
740 check caterpillar position
750 check keyboard
760-800 check for movement
810-1120 define new screen positions
1150-1160 check new screen positions
1170-1180 display caterpillar in new positions
1190-1220 check new screen positions
1230-1300 you've eaten a leaf
1310-1580 a patch is completed
1590-1650 you bumped into your own body
1660-1700 you hit a mushroom
1710-1800 you went into the hedge
1810-1840 display number of remaining lives
1850-1930 have you run out of lives?
1940-2200 end of game
2210-2240 display score and screen messages
2250-2370 display instructions

mushroom patch. If you touch it the game ends.

Your first patch has 20 mushrooms and 10 leaves. Once you finish a patch, you move on to a new one with five more mushrooms and one more leaf. This makes each patch harder than the one before.

Control the caterpillar with the four cursor keys, keeping the alpha-lock button down. If you find the game too fast, stop the caterpillar moving by holding down any other key apart from the cursor keys.

Your score, patch number and number of lives are constantly displayed at the top of the screen. You score 10 points for each leaf you eat, and whenever you complete a patch you get a bonus score. The bonus depends on which patch you've completed and how many of your lives are left.

At the beginning of the game simple instructions are displayed, but if you decide to play again at the end of a game, the instructions won't be repeated.

Variables

NOM number of mushrooms
NOL number of leaves
PAT patch number
LI number of lives
SC score
Q detect position of caterpillar
COT change text to white
J,F,A,B,O,P used to move caterpillar
LEAF number of leaves eaten
CHAR shape of caterpillars head
MS hold screen messages and score
MD/MA positions of MS
MUSH display required number of mushrooms
Y/X positions of leaves and mushrooms
WM1 check positions of mushrooms tops
WM2 check positions of mushrooms stalks
LEA display required number of leaves
WAL check positions of leaves
NOCD clear screen
DELAY produce short delay
NU change colour of caterpillar's body
V change volume of sound
NUCD clear screen
SO change musical notes
I display screen messages

The more more you

```

1 REM *****
2 REM *CATERPILLAR CRAWL*
3 REM *      BY      *
4 REM *  STUART LENKER  *
5 REM * 25TH APRIL 1984 *
6 REM *                *
7 REM * HIGH-SCORE 2870 *
8 REM *  BY JULIETTE   *
9 REM *****
10 GOSUB 2250
15 NOM=20
20 NOL=10
30 PAT=1
40 LI=3
50 SC=0
60 Q=32
70 CALL CLEAR
80 CALL SCREEN(2)
90 FOR COT=1 TO 8
100 CALL COLOR(COT,16,1)
110 NEXT COT
120 CALL COLOR(13,13,1)
130 CALL COLOR(14,16,1)
140 CALL COLOR(15,7,1)
150 CALL COLOR(16,11,1)
160 CALL CHAR(128,"AA55AA55AA55A
A55")
170 CALL CHAR(129,"081C3E7F7F3E0
808")
180 CALL CHAR(136,"18183C3C7E")
190 CALL CHAR(144,"000000183C7EF
FFF")
200 CALL CHAR(152,"3C7EFFFFFFFFF7
E3C")
210 CALL CHAR(153,"814224183C7E7
E3C")
220 CALL CHAR(154,"3C7E7E3C18244
281")
230 CALL CHAR(155,"0162F4F8F8F46
201")
240 CALL CHAR(156,"80462F1F1F2F4
680")
250 J=0
260 F=1
270 A=4
280 B=2
290 O=1

```

Watch yourself expand as you gobble up more and more in this caterpillar game by Stuart Lenker

you eat, the grow

```

300 P=0
310 LEAF=1
320 CHAR=154
330 CALL HCHAR(2,1,128,32)
340 CALL HCHAR(24,1,128,32)
350 CALL VCHAR(2,1,128,23)
360 CALL VCHAR(2,32,128,23)
370 M$="SCORE-"&STR$(SC)
380 MD=1
390 MA=1
400 GOSUB 2210
410 M$="PATCH-"&STR$(PAT)
420 MD=1
430 MA=13
440 GOSUB 2210
450 M$="LIVES-"&STR$(LI)
460 MD=1
470 MA=23
480 GOSUB 2210
490 FOR MUSH=1 TO NOM
500 RANDOMIZE
510 X=INT(31*RND)+1
520 IF X<3 THEN 510
530 Y=INT(22*RND)+1
540 IF Y=1 THEN 530
550 CALL GCHAR(Y,X,WM1)
560 IF WM1<>32 THEN 500
570 CALL GCHAR(Y+1,X,WM2)
580 IF WM2<>32 THEN 500
590 CALL HCHAR(Y,X,144)
600 CALL HCHAR(Y+1,X,136)
610 NEXT MUSH
620 FOR LEA=1 TO NOL
630 RANDOMIZE
640 X=INT(31*RND)+1
650 IF X<3 THEN 640
660 Y=INT(23*RND)+1
670 IF Y=1 THEN 660
680 CALL GCHAR(Y,X,WAL)
690 IF WAL<>32 THEN 630
700 CALL HCHAR(Y,X,129)
710 NEXT LEA
720 CALL HCHAR(A,B,CHAR)
730 CALL HCHAR(A-F,B-J,152)
740 IF Q<>32 THEN 1190
750 CALL KEY(O,X,Y)
760 IF Y=0 THEN 1130

```

```

770 IF X=83 THEN 810
780 IF X=68 THEN 890
790 IF X=69 THEN 970
800 IF X=88 THEN 1050 ELSE 720
810 B=B-1
820 F=0
830 J=-1
840 P=-1
850 D=0
860 CHAR=156
870 CALL GCHAR(A,B,Q)
880 GOTO 720
890 B=B+1
900 F=0
910 J=1
920 P=1
930 D=0

```

Hints on conversion

If you are going to try and convert this program to another machine, then remember the following points:

- 1 The Texas has a 24 by 32 graphic screen.
- 2 It has a 24 by 28 text screen.
- 3 It uses user-defined graphics.

A good amount of space can be saved by using multiple line statements. Below are the more unusual commands:

CALL CHAR (ASCII code number, hexadecimal string) Defines characters used in the program. The hexadecimal string describes the pattern to be given to the character with ASCII code stated.

CALL HCHAR (row number, column number, ASCII code, number of repetitions) Places character on screen at row and column specified, and repeats it horizontally the number of times stated. If this fourth value is omitted then the character is displayed only once. The character used is the one whose ASCII code appears in the command.

CALL VCHAR (row number, column number, ASCII code, number of repetitions) Works the same way as **CALL HCHAR** only repeats vertically instead of horizontally.

CALL GCHAR (row number, column number, numerical variable) Equivalent of **PEEK**; assigns ASCII code of character at row and column stated to numerical variable.

CALL CLEAR Clears the screen. Equivalent to **CLS**.

CALL SCREEN (colour code) Specifies screen colour, using colour code.

CALL COLOUR (character set number, foreground colour code, background colour code) Specifies foreground and background colours for all characters in set stated.

CALL SOUND (duration, frequency, volume) Produces sound with duration, frequency, and volume specified. Duration is in milliseconds, frequency in Hertz and volume in scale from 0 (loudest) to 30 (quietest).

CALL KEY (key unit, k,s) Equivalent of **INKEY\$** or **GET**. Returns code if key pressed in variable k.

These are the 16 colours:

- 1 transparent
- 2 black
- 3 medium green
- 4 light green
- 5 dark blue
- 6 light blue
- 7 dark red
- 8 Cyan
- 9 medium red
- 10 light red
- 11 dark yellow
- 12 light yellow
- 13 dark green
- 14 magenta
- 15 grey
- 16 white


```

940 CHAR=155
950 CALL GCHAR(A,B,Q)
960 GOTO 720
970 A=A-1
980 F=-1
990 J=0
1000 P=0
1010 D=-1
1020 CHAR=153
1030 CALL GCHAR(A,B,Q)
1040 GOTO 720
1050 A=A+1
1060 F=1
1070 J=0
1080 P=0
1090 D=1
1100 CHAR=154
1110 CALL GCHAR(A,B,Q)
1120 GOTO 720
1130 B=B+P
1140 A=A+D
1150 CALL GCHAR(A,B,Q)
1160 IF Q<>32 THEN 1170 ELSE 720
1170 CALL HCHAR(A,B,CHAR)
1180 CALL HCHAR(A-F,B-J,152)
1190 IF Q=129 THEN 1230
1200 IF Q=128 THEN 1710
1210 IF Q=152 THEN 1590
1220 IF (Q=136)+(Q=144) THEN 1660
1230 CALL SOUND(99,-1,2,320,0,55
0,3)
1240 SC=SC+10
1250 LEAF=LEAF+1
1260 Q=32
1270 M$=STR$(SC)
1280 MD=1
1290 MA=7
1300 GOSUB 2210
1310 IF LEAF=NOL+1 THEN 1320 ELS
E 720
1320 CALL SOUND(100,440,0,540,2)
1330 CALL SOUND(100,550,0,650,2)
1340 CALL SOUND(100,440,0,540,2)
1350 CALL SOUND(100,660,0,760,2)
1360 CALL SOUND(100,880,0,980,2)
1370 FOR NU=3 TO 23
1380 CALL HCHAR(NU,2,32,30)
1390 NEXT NU
1400 M$="WELL DONE-PATCH "&STR$(
PAT)&" COMPLETED"
1410 MD=5
1420 MA=2
1430 GOSUB 2210
1440 M$="BONUS OF "&STR$(PAT*LI*
10)
1450 MD=7

```

```

1460 MA=10
1470 GOSUB 2210
1480 SC=SC+(PAT*LI*10)
1490 M$=STR$(SC)
1500 MD=1
1510 MA=7
1520 GOSUB 2210
1530 PAT=PAT+1
1540 NOL=9+PAT
1550 NOM=NOM+5
1560 FOR DELAY=1 TO 500
1570 NEXT DELAY
1580 GOTO 70
1590 CALL SOUND(600,110,30,110,3
0,500,30,-8,0)
1600 FOR NU=1 TO 6
1610 CALL COLOR(16,7,1)
1620 CALL COLOR(16,11,1)
1630 NEXT NU
1640 LI=LI-1
1650 GOTO 1810
1660 FOR V=0 TO 30 STEP 2
1670 CALL SOUND(-99,-6,V)
1680 NEXT V
1690 LI=LI-1
1700 GOTO 1810
1710 FOR V=30 TO 0 STEP -1
1720 CALL SOUND(-250,-2,V)
1730 NEXT V
1740 FOR V=0 TO 30
1750 CALL SOUND(-500,-2,V)
1760 NEXT V
1770 FOR NU=3 TO 23
1780 CALL HCHAR(NU,2,32,30)
1790 NEXT NU
1800 GOTO 2050
1810 M$=STR$(LI)
1820 MD=1
1830 MA=29
1840 GOSUB 2210
1850 IF LI<1 THEN 1940
1860 D=0
1870 P=0
1880 Q=32
1890 CALL HCHAR(A,B,CHAR)
1900 CALL HCHAR(A-F,B-J,152)
1910 CALL KEY(0,K,S)
1920 IF S=0 THEN 1910
1930 GOTO 720
1940 FOR SD=150 TO 110 STEP -1
1950 CALL SOUND(-100,SD,0)
1960 NEXT SD
1970 FOR NU=3 TO 23
1980 CALL HCHAR(NU,2,32,30)
1990 NEXT NU
2050 M$="GAME OVER"
2060 MD=9

```


PROGRAM

```

2070 MA=11
2080 GOSUB 2210
2090 M$="YOUR FINAL SCORE IS "&S
TR$(SC)
2100 MD=11
2110 MA=5
2120 GOSUB 2210
2130 M$="PRESS Y TO PLAY AGAIN"
2140 MD=16
2150 MA=5
2160 GOSUB 2210
2170 CALL KEY(0,K,S)
2180 IF S=0 THEN 2170
2190 IF K=89 THEN 15 ELSE 2200
2200 END
2210 FOR I=1 TO LEN(M$)
2220 CALL HCHAR(MD,MA+I,ASC(SEG$(M$,I,1)))
2230 NEXT I
2240 RETURN
2250 CALL CLEAR
2260 CALL SCREEN(2)
2270 FOR COT=1 TO 8
2280 CALL COLOR(COT,16,1)
2290 NEXT COT
2300 PRINT TAB(5);"CATERPILLAR C
RAWL";TAB(5);"*****"
::::
2310 PRINT "MOVE THE CATERPILLAR
AROUND"::"THE MUSHROOM PATCH,WH
ILE":
2320 PRINT "EATING THE LEAVES AN
D"::"DODGING THE MUSHROOMS.":
2330 PRINT "AVOID THE SURROUNDI
NG HEDGE,"::"AND YOUR OWN BODY.":
::
2340 PRINT "PRESS ANY KEY TO CON
TINUE"
2350 CALL KEY(0,K,S)
2360 IF S=0 THEN 2350
2370 RETURN

```



MEGASAVE

SPECTRUM

Sabre Wulf	£8.25
Psytron	£6.30
Mugsy	£5.95
Jack & Beanstalk	£4.95
Fighter Pilot	£6.25
Tornado Low Level	£4.75
Hulk	£8.50
Lords of Midnight	£8.50
Kosmic Kanga	£8.25
Jet Set Willy	£4.75
Beaky	£4.85
Ad Astra	£4.95
Night Gunner	£5.50
Match Point	£6.85
Trashman	£4.75
Moon Alert	£4.90
Antics	£5.75
Wars of the World	£6.75
Les Flies	£5.50
Cavelon	£4.95

FANTASTIC SAVINGS

SPECTRUM

Atic Atac	£4.75
Automania	£5.75
Blue Thunder	£4.75
Inferno	£5.50
Code Name MAT	£5.20
Millionaire	£4.50
Full Throttle	£6.95
Sink the Titanic	£6.75
Specgraph	£8.25
Cosmic Cruiser	£4.25
World Cup F'ball	£5.95
Factory Brakeout	£5.95
Quill	£10.75
WTH at Sea	£5.95
Blade Alley	£4.75
3D Luna Attack	£4.75
Caesar the Cat	£5.95
Pyramid	£4.95
Android II	£4.75
Air Traffic Control	£4.95

COMMODORE

Son of Bagger	£6.50
Beach Head	£8.50
Valhalla	£11.50
Quill	£11.50
Sheep in Space	£6.40
Gyropod	£5.50
Loco	£6.50
Hulk	£8.50
Arabian Knights	£5.85
Solo Flight	£12.00
Trash Man	£6.85
Star Trooper	£5.75
Cavelon	£5.50
Encounter	£8.75
House of Usher	£5.85
Gilligan's Gold	£6.90
Blue Thunder	£5.95
Android II	£5.85
All £7 Interceptor	£5.75
All £7.95 Anirog	£6.30

FREE POST PLEASE STATE WHICH MICRO.

SEND CHEQUE/PO TO:

MEGASAVE, DEPT 2, 76 WESTBOURNE TERRACE, LONDON W2

PROGRAM FOR PROFIT WITH YOUR SPECTRUM

Software has now outstripped the areas of basic and simple machine code programs. Do you have the ideas for games but not the know how to produce the code required? Are you put off by the complexities of writing complicated routines?

Have you the problem of finding the time to write that great adventure or arcade game? Professional software houses use special programs to help them generate their product. Now there is available to you a package of commercially marketed programs with which you can write your own text adventures, graphics adventures and arcade games. Packaged together with two books on arcade and adventure programming and sent to you with a Free copy of our fact sheets "Programming for the games market", which includes many hints and tips and a Softwarehouse directory, you can now compete with the top games writers. (Programs include The Quill, Dungeon Builder, White Lightning.) Purchased separately these items would cost you £49.75. As a special offer we are packaging the items at £42.75 a saving of £7.00.

Send for your pack or send SAE for further details to

RAMTOPS, 47 BROOM LAND, LEVENSHULME, MANCHESTER M19 2TX



LES FLIES

ARCADE ADVENTURE
FOR SPECTRUM 48K

£6.95

You control that Pink character with the tail in your quest for the Purple Puma, a priceless gemstone, avoid the Police cars and enter the buildings, evading Gend'armerie Kaolin (disguised as a chef) and of course Claudeau himself.

Pick up the items as you find them, we leave you to find out what they do!

Bon Chance

PSS SOFTWARE

452 STONEY STANTON RD. COVENTRY CV6 5DG. TEL: COVENTRY (0203) 667556



TI-99/4A LANTERN TI-99/4A

SAVE £££'s ON THESE SUPERB GAMES

£ Hunchback Havock (BASIC) £5.95	£ Daddie's Hot Rod (BASIC) £5.95
£ Arcade action in TI-BASIC! Race with Egor thru four castles and 24 different sheets. Superb graphics and great fun.	£ Drive through sheep fields, woods, over rivers and avoid all obstacles, then pick up hitch-hikers down perilously twisting roads.
£ The Black Tower (BASIC) £5.95	£ Builder/Minefield (BASIC) £4.95
£ Unlock the mysteries of the black tower and rescue the Lady Gwen in our latest full 16K adventure. Addictive and entertaining.	£ Two great games in one. Builder, drive a truck, pick up bricks and build a wall but watch the obstacles. Minefield, drive a tank through a minefield while being shelled.
£ Runner On Treiton (BASIC) £5.95	£ Battlestar Attack (EXT-BASIC) £6.95
£ You are on the planet Treiton your task is to collect jewels. This semi-text, semi-graphics adventure requires skill and a good memory.	£ Stop the battlestar before it blows up your home planet. A fast game with excellent graphics. HCW said: 'even at level one a very addictive game.'

Special Offer

Mention this ad when ordering any of the above games and you will ONLY pay £4.95. per title!

Don't delay — order Now! Offer ends 31st July.

Send cheque or P.O. to

LANTERN SOFTWARE,

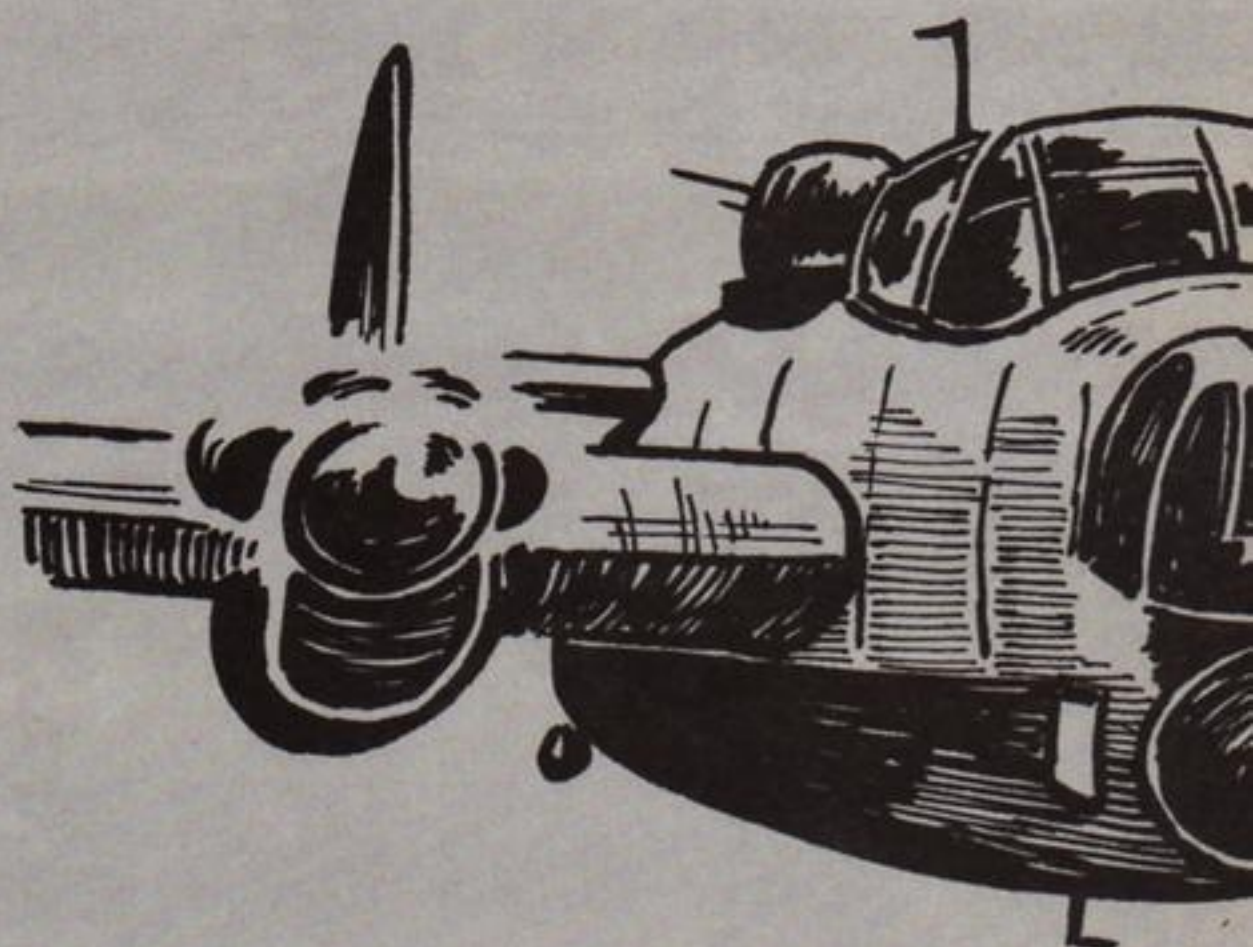
4 Haffenden Road, Tenterden, Kent TN30 6QD

or SAE for full list. Look out for our products at your TI retailer.


```

100 CALL CLEAR
110 CALL SCREEN(2)
120 CALL CHAR(100,"3C7EFFFFFFFFF7
E3C")
130 CALL CHAR(104,"0800080008000
8")
140 CALL CHAR(112,"00001818FFFFF
F")
150 CALL CHAR(113,"000020702")
160 CALL CHAR(128,"00001818FFFFF
F")
170 CALL CHAR(136,"A963F33F9F1F7
FFF")
180 CALL CHAR(137,"95C6CFFCF9F8F
EFF")
190 CALL CHAR(138,"FF7F1F9F3FF36
3A9")
200 CALL CHAR(139,"FFFEF8F9FCCFC
695")
210 CALL CHAR(144,"3C3C3C3C3C3C3
C3C")
220 FOR I=2 TO 8
230 CALL COLOR(I,11,1)
240 NEXT I
250 PRINT TAB(10);"MINEFIELD": "
*****": "DO YOU WANT INSTRUCTIONS": "(Y
OR N)": "
260 CALL KEY(O,K,S)
270 IF S=0 THEN 260
280 IF (K=78)+(K=110) THEN 680
290 IF (K=89)+(K=121) THEN 310
300 GOTO 250
310 RESTORE 3000
320 CALL CLEAR
330 FOR I=1 TO 10
340 READ A$,B,C
350 GOSUB 2450
360 NEXT I
370 FOR DE=1 TO 400
380 NEXT DE
390 CALL CLEAR
400 RESTORE 3030
410 FOR I=1 TO 5
420 READ A$,B,C
430 GOSUB 2450
440 NEXT I
450 GOTO 720
460 CALL CLEAR
470 A$="HIGH SCORES"
480 B=1
490 C=10
500 GOSUB 2450
510 A$="-----"
520 B=2
530 C=9
540 GOSUB 2450

```



If you know and love the game Bomber, you'll drool over this. It's an adaptation of that well known game, but this time you start from the bottom and work your way up.

This game features a mystery bonus which can double your score, 18 levels of play, with increasing difficulty, a mystery blowing-up of the submarine and an end of sheet bonus. Once you have reached 10,000 points you get an extra submarine as a reward for your efforts. The hall of fame records your high scores.

Texas BASIC differs from other BASICs in that you don't have direct access to the processor. Instead you use built-in machine code subprograms for sound and graphics. Only statement lines are supported.

Hints on conversion

Clear clear screen
Color (a,b,c) set colour of character; a foreground, b colour, c background colour
Char (a,a\$) redefine character a to pixel pattern defined by a\$
HCHAR (a,b,c,(d)) place character c at row a and column b, optionally repeat d times
VCHAR (a,b,c,(d)) place character c at row a and column b, optionally repeat d times vertically
Sound (a,b,c) play frequency b for a milliseconds at volume c
GCHAR (a,b,c) return the ASCII code into variable c of the character at screen row a, screen column b
Key (0,a,b) return ASCII code into a of key pressed. Return 0 into b if no key pressed

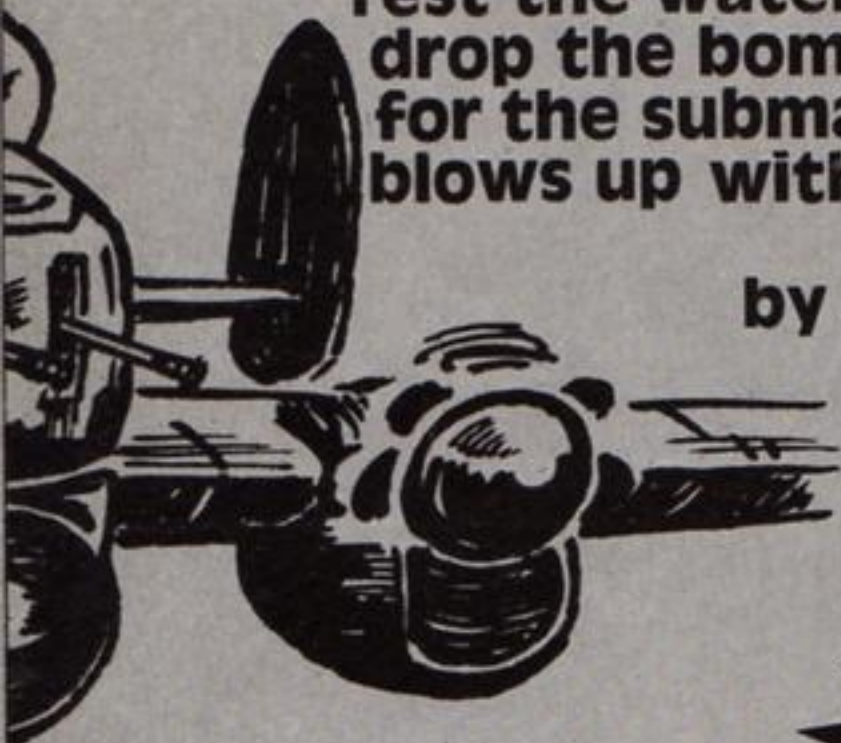
Variables

AS get string from data statements
B character row in print routine
C character column in print routine
D skill level
E lives
F submarine column
G bomb character
H submarine row
M depth of mine
N volume
O second rest variable
P bomb row
Q bomb column
R first test variable
T character
V hit
X third rest

How it works

100-110 set screen colour
120-210 define characters
220-240 set colours
250-300 test for instructions
310-450 instructions routine
460-740 high score hall of fame
750-1390 set up screen
1400-1470 flashing mines and start
1480-1710 main game sequence
1720-1980 end of sheet bonus
1990-2020 test for bonus sub
2030-2180 test bomb for hit and bonus
2190-2250 move bomb
2260-2440 hit mine routine
2450-2490 print at routine
2500-2600 blow up sub routine
2610-2700 game over
2710-2750 test for high scores
2760-2840 set up high score message
2850-2990 sort high scores
3000-3060 a data for print messages

Test the waters before you drop the bomb. Watch out for the submarine which blows up without warning, in this game by Jon Dempsey



Bom a w a y !

```

700 C=2
710 GOSUB 2450
720 CALL KEY(0,K,S)
730 CALL SOUND(-99,-4,0)
740 IF S=0 THEN 720
750 CALL CLEAR
760 CALL COLOR(9,10,1)
770 CALL COLOR(10,16,1)
780 CALL COLOR(11,11,1)
790 CALL COLOR(12,13,13)
800 CALL COLOR(13,2,13)
810 CALL COLOR(14,9,1)
820 CALL COLOR(15,14,13)
830 CALL COLOR(16,1,1)
840 FOR I=2 TO 8
850 CALL COLOR(I,2,16)
860 NEXT I
870 CALL HCHAR(1,1,120,96)
880 CALL HCHAR(23,1,120,64)
890 A=0
900 A$="POINTS="&STR$(A)&"xxxx"
910 B=2
920 C=16
930 GOSUB 2450
940 X=1
950 D=6
960 E=3
970 F=2
    
```

```

550 FOR I=10 TO 2 STEP -2
560 A$=STR$(I/2)&". "&CHR$(32)&STR$(V(I/2))
570 B=(I*2)+2
580 C=3
590 GOSUB 2450
600 NEXT I
610 FOR I=10 TO 2 STEP -2
620 A$="__"&V$(I/2)
630 B=(I*2)+2
640 C=14
650 GOSUB 2450
660 CALL SOUND(-99,110*I,4)
670 NEXT I
680 A$="PRESS ANY KEY TO PLAY"
690 B=24
    
```



```

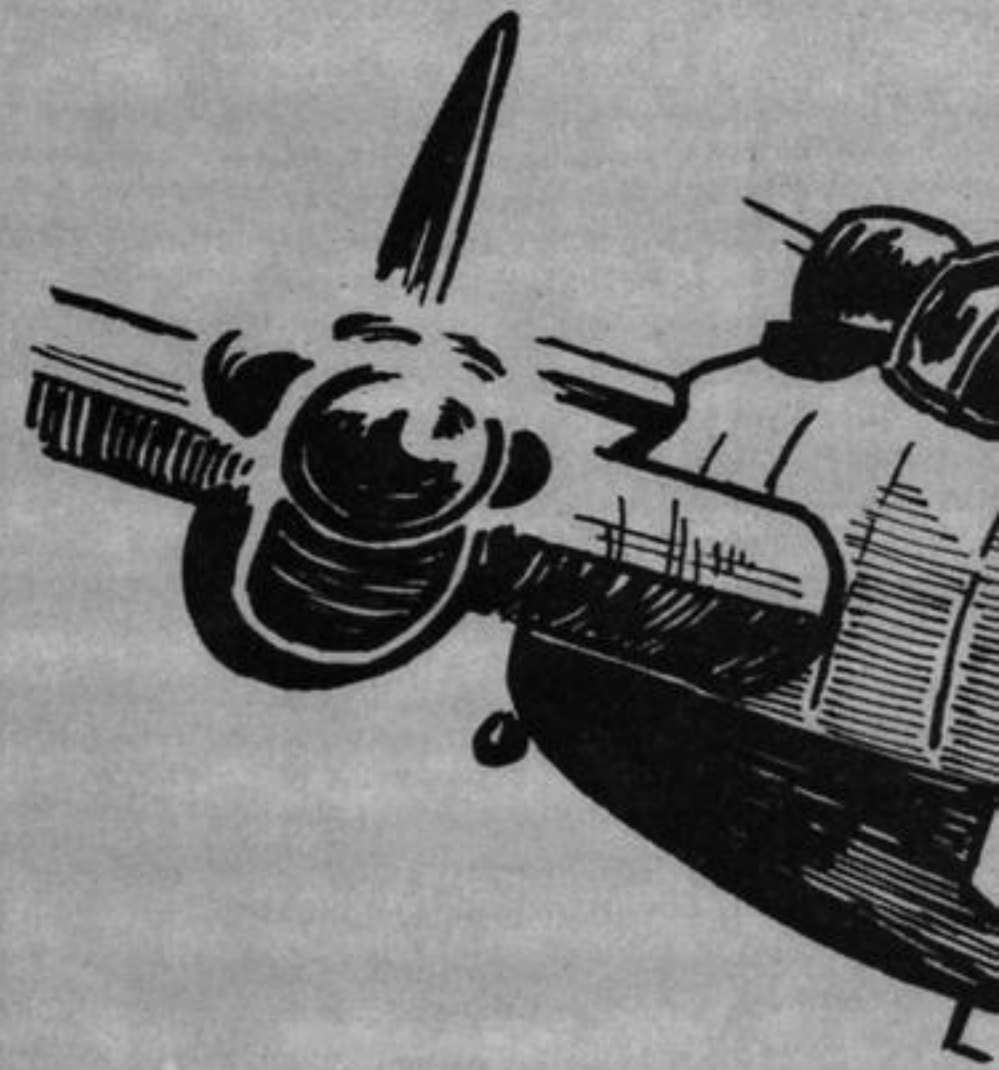
980 G=18
990 A$="SHOTS="&STR$(G)&"x"
1000 B=2
1010 C=3
1020 GOSUB 2450
1030 IF D=18 THEN 1060
1040 H=D+4
1050 GOTO 1070
1060 H=22
1070 RANDOMIZE
1080 CALL HCHAR(4,1,32,608)
1090 CALL HCHAR(3,1,120,32)
1100 CALL HCHAR(23,1,120,32)
1110 J=INT(RND*15)+4
1120 L=INT(RND*32)+1
1130 Y=INT(15*RND)+4
1140 YY=INT(L*RND)+1
1150 CALL HCHAR(J,L,159)
1160 CALL HCHAR(Y,YY,158)
1170 FOR I=5 TO 29 STEP 2
1180 M=INT(RND*D)+1
1190 CALL VCHAR(4,I,104,M)
1200 CALL HCHAR(M+3,I,100)
1210 CALL SOUND(-50,-5,0)
1220 NEXT I
1230 D=D+1
1240 IF D<>18 THEN 1270
1250 Z=D+1
1260 GOTO 1280
1270 Z=D
1280 A$="CARD="&STR$(Z-6)&"x"
1290 B=24
1300 C=3
1310 GOSUB 2450
1320 CALL HCHAR(24,20,120,8)
1330 GOSUB 1350
1340 GOTO 1400
1350 FOR I=1 TO ((E-1)*2)STEP 2
1360 CALL SOUND(-10,-1,0)
1370 CALL HCHAR(24,20+I,128)
1380 NEXT I
1390 RETURN
1400 CALL KEY(0,K,S)
1410 CALL COLOR(9,2,1)
1420 CALL HCHAR(H,F,112)
1430 CALL HCHAR(H,F,32)
1440 CALL SOUND(-200,500,5,-3,5,
1000,10)
1450 CALL COLOR(9,10,1)
1460 U=0
1470 IF S=0 THEN 1400
1480 CALL HCHAR(H,F,112)
1490 CALL KEY(0,K,S)
1500 IF S=0 THEN 1530
1510 IF G=0 THEN 2500
1520 IF K=32 THEN 2030
1530 CALL GCHAR(H,F+1,D)

```

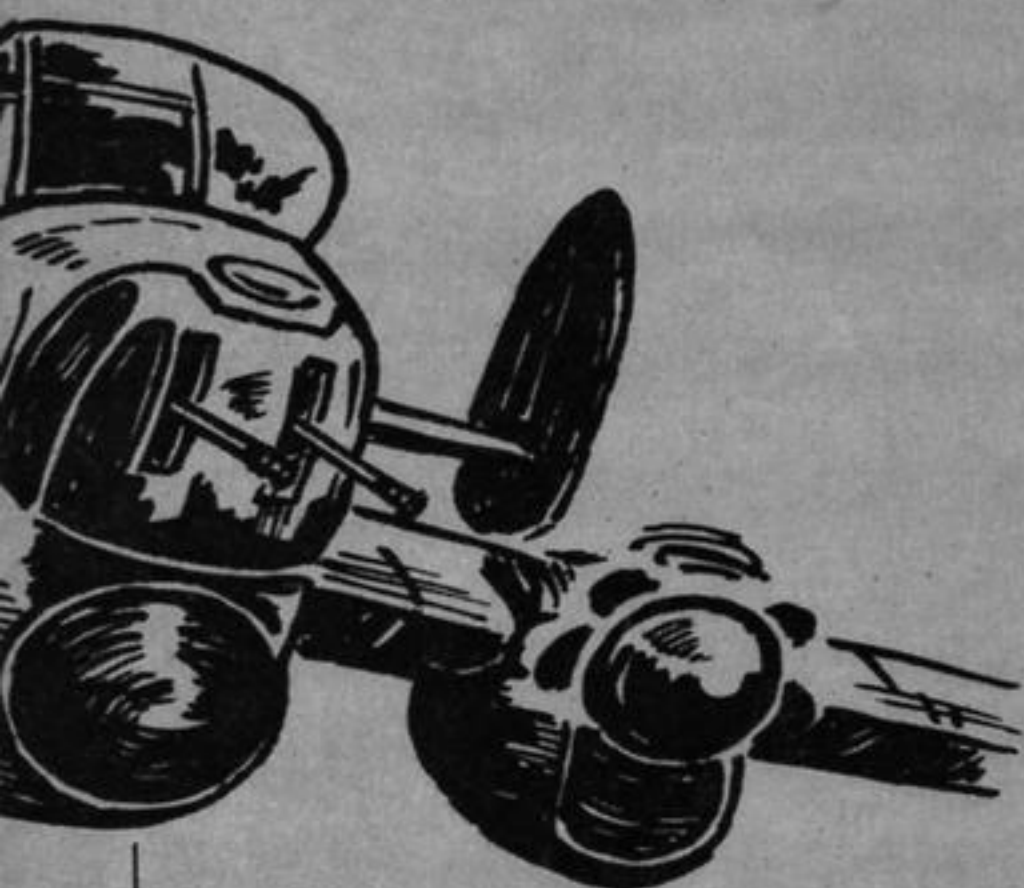
```

1540 IF (D=100)+(D=158)THEN 2500
1550 F=F+1
1560 IF F<32 THEN 1640
1570 CALL HCHAR(H,F,32)
1580 F=2
1590 H=H-1
1600 A$=STR$(G)&"x"
1610 B=2
1620 C=9
1630 GOSUB 2450
1640 CALL VCHAR(4,31,32,19)
1650 CALL HCHAR(H,F-1,32)
1660 IF H>3 THEN 1480
1670 IF D>=18 THEN 1680 ELSE 170
0
1680 D=18
1690 GOTO 980
1700 CALL HCHAR(4,1,32,32)
1710 IF G=0 THEN 970
1720 A$="BONUS"
1730 B=10
1740 C=10
1750 GOSUB 2450
1760 IF G>1 THEN 1820
1770 A$=STR$(G)&" BOMB"
1780 B=12
1790 C=10
1800 GOSUB 2450
1810 GOTO 1860
1820 A$=STR$(G)&" BOMBS"
1830 B=12
1840 C=10

```



PROGRAM



```

1850 GOSUB 2450
1860 A$="@ 50 POINTS EACH"
1870 B=14
1880 C=10
1890 GOSUB 2450
1900 A$=STR$(G*50)&" POINTS"
1910 B=16
1920 C=10
1930 GOSUB 2450
1940 A=A+(G*50)
1950 A$=STR$(A)&"x"
1960 B=2
1970 C=23
1980 GOSUB 2450
1990 IF (A>=10000)*(X=1) THEN 200
0 ELSE 970
2000 E=E+1
2010 X=X+1
2020 GOTO 970
2030 P=H
2040 G=G-1
2050 CALL SOUND(-100,-5,4)
2060 Q=F
2070 Q=Q+1
2080 P=P-1
2090 CALL GCHAR(P,Q,R)
2100 CALL GCHAR(H,F+1,D)
2110 IF D=100 THEN 2300
2120 IF (R<>159)*(R<>32)*(R<>100)
)*(R<>158) THEN 1480
2130 IF (R=32)+(R=158) THEN 2190
2140 IF R=100 THEN 2300

```

```

2150 A=A*2
2160 CALL SOUND(-150,1000,3,-7,1
0,330,3)
2170 CALL HCHAR(P,Q,32)
2180 GOTO 2340
2190 CALL HCHAR(P,Q,113)
2200 CALL SOUND(-50,3000,7)
2210 CALL HCHAR(H,F,112)
2220 IF F=32 THEN 1570
2230 IF Q<32 THEN 2260
2240 CALL HCHAR(P,Q,32)
2250 GOTO 1570
2260 CALL HCHAR(P,Q,32)
2270 CALL HCHAR(H,F,32)
2280 F=F+1
2290 GOTO 2070
2300 U=U+1
2310 CALL VCHAR(4,Q,32,P-3)
2320 A=A+50
2330 CALL SOUND(-750,110,2,-7,0,
200,5)
2340 IF (A>=10000)*(X=1) THEN 235
0 ELSE 2390
2350 E=E+1
2360 X=X+1
2370 GOSUB 1350
2380 CALL SOUND(-750,110,2,-6,0,
192,2)
2390 A$=STR$(A)&"x"
2400 B=2
2410 C=23
2420 GOSUB 2450
2430 IF U=13 THEN 1710
2440 GOTO 1480
2450 FOR T=1 TO LEN(A$)
2460 Z=ASC(SEG$(A$,T,1))
2470 CALL HCHAR(B,T+C,Z)
2480 NEXT T
2490 RETURN
2500 CALL HCHAR(H,F,136)
2510 CALL HCHAR(H,F+1,137)
2520 CALL HCHAR(H+1,F,138)
2530 CALL HCHAR(H+1,F+1,139)
2540 FOR I=0 TO 30
2550 CALL SOUND(-200,-7,I,110,I)
2560 NEXT I
2570 E=E-1
2580 D=D-1
2590 IF E=0 THEN 2610
2600 GOTO 970
2610 A$=STR$(G)&"x"
2620 C=9
2630 B=2
2640 GOSUB 2450
2650 A$="GAME OVER"
2660 B=12
2670 C=10

```


TI-99/4A PROGRAM

```

2680 GOSUB 2450
2690 FOR DE=1 TO 700
2700 NEXT DE
2710 CALL CLEAR
2720 FOR I=2 TO 8
2730 CALL COLOR(I,7,1)
2740 NEXT I
2750 IF A>V(5) THEN 2760 ELSE 460
2760 V(5)=A
2770 RESTORE 3050
2780 FOR I=1 TO 6
2790 READ A$,C
2800 B=10+(I*2)
2810 GOSUB 2450
2820 NEXT I
2830 INPUT W$
2840 IF LEN(W$)>10 THEN 2830
2850 V$(5)=W$
2860 CALL CLEAR
2870 FOR II=1 TO 5
2880 FOR I=1 TO II
2890 IF V(II)<V(I) THEN 2970
2900 CALL SOUND(-99,1100,I,2000,
II)
2910 VV=V(I)
2920 VV$=V$(I)

```

```

2930 V(I)=V(II)
2940 V$(I)=V$(II)
2950 V(II)=VV
2960 V$(II)=VV$
2970 NEXT I
2980 NEXT II
2990 GOTO 460
3000 DATA *****,1,10,MINEF
IELD,3,11,*****,5,10,YOUR
SUB HAS RUN OUT OF AIR,7,3
3010 DATA AND YOU MUST SURFACE--
IN,9,2,A MINEFIELD.,11,2,YOU HAV
E EIGHTEEN MISSILES,14,3
3020 DATA TO HIT ALL THE MINES.,
16,2,YOU GET AN EXTRA SUB AT,19,
3,10000 POINTS.,21,2
3030 DATA THERE IS A MYSTERY BON
US,5,3,WHICH DOUBLES YOUR SCORE.
,7,2,USE THE SPACE BAR TO FIRE.,
10,3
3040 DATA GOOD LUCK.,15,10,PRESS
ANY KEY TO PLAY.,24,2
3050 DATA CONGRATULATIONS,4,YOUR
SCORE IS,4,ONE OF THE,4,FIVE HI
GHEST,4,PLEASE ENTER,4
3060 DATA YOUR NAME,4

```

KFTmarketing

Regent Works, Regent Street, Kimberley, Nottingham, NG16 2LW. Telephone (0602) 383931

Give your home computer the professional look with

The NEON  GLARE FILTER

Filters out the harsh glare from T.V. screens

Reduces eye strain and sore eyes caused by
prolonged programming

Converts any blk/white or colour T.V. screen into a
green V.D.U. type display

Fitted or removed in seconds

Virtually INDESTRUCTABLE and complete with
superior quality case and simple instructions

99p incl. p&p.

cheques & P.O. payable to K.F.T. Marketing

Name _____

Address _____

Post Code _____

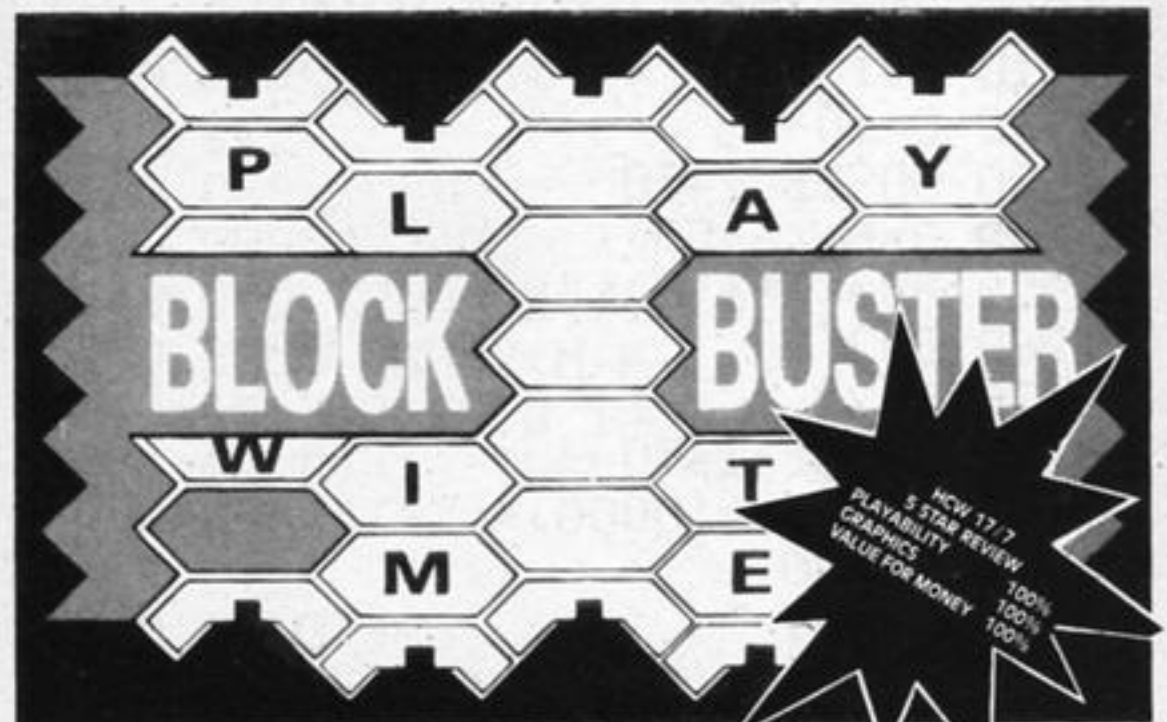
Trade Enquiries welcome

K.F.T. Marketing

Regent Works, Regent St,
Kimberley, Notts NG16 2LW

Tel: 0602 383931

ADAPTED FROM THE POPULAR
T.V. SERIES!!!



BLOCK-BUSTER is an exciting quiz game for ONE or TWO players and will appeal to a wide range of ages and skills with nine levels of play against the clock. The game tests speed of recall involving anticipation of the question as it 'unfolds' across the screen and speed of response dependant on the level of play.

BLOCK BUSTER is supplied with the FIRST ADDITIONAL QUESTION TAPE FREE to maintain the challenge of the game. Now available at good computer shops or direct with cheque/P.O. for £5.95 inclusive. ACCESS and VISA orders taken on our 24 Hour Hotline.

COMPUSOUND

DEPT HCW, 32-33 LANGLEY CLOSE, REDDITCH, WORCS B98 0ET
TELEPHONE (0527) 21429 (21439 24 Hr. HOTLINE)

**Cathy Oxspring, Paignton,
Devon**

Solve those quests

**John Rundle, Aldershot,
Hants**

Dive for pearls

```
1645 IF P = P + 10 THEN
```

Maurice Rymill, Birmingham

Live on, Manic Miner

SYS 16384

**M. Livingstone, Doncaster,
S Yorks**

Sord support

If he finds his local dealer doesn't provide the software he needs, he should write to CGL. I have written to them many times and I have nothing but praise for their help.

We do feature Sord programs when we have good listings to publish. If you have excellent games you have written, send them in for publication.

60 Progs
reprint 1

**D.J. Marshall, Northolt,
Middx**

60 Progs
reprint 2

He was also told that if he forwarded the first page of his book to the address shown on that page he would receive one of the corrected manuscripts. This we did, and since receiving our new books we have had few, if any, problems. I suggest that R.T. Goodwin does the same and obtains a

Texas fan

Novice needs advice

When I run out of listings I like to sit down and write my text adventure. I'm

Software winner

Page 16 HOME COMPUTING WEEKLY 31 July 1984

COMPETITION

Intrigue your friends with an Intrigue T-shirt.

We've got 75 to give away to each of our prizewinners — and there's £40-worth of great Intrigue games as well in the first prize.

Intrigue Software, run by father and son team Dennis and Martin Webb, launched in November and now has a range of graphic arcade adventures for the TI-99/4A.

The theme of our spot the difference cartoon is taken from Intrigue's latest release, Atlantis.

Set undersea, it features Lionel, who is the star of other Intrigue games. It costs £6.95 and runs on the standard TI-99/4A.

Another TI title, Snout of Spout, is due out soon.

And, now Intrigue has a team of programmers, the firm is branching out into Commodore 64 software. The first title, under wraps at present, is a new game which will be on sale in time for Christmas.

Our first prizewinner will receive from Intrigue a T-shirt bearing the firm's distinctive logo, plus one each of all Intrigue's games.

They are: Mania, Adventuremania and Beneath the Stars, all £5.95; Lionel and the Ladders and Lionel and the Castle, both £7.95; and Atlantis, £6.95. Lionel and the Ladders and Lionel and the Castle need the TI's Extended BASIC module. All the others run on the standard TI.

The 74 runners-up will receive Intrigue T-shirts.

The prizes will arrive from Intrigue within 28 days of the results appearing in Home Computing Weekly.

How to enter

Study the two cartoons — there are a number of differences between

Intrigue T-shirts and software to be won

**We're offering
both software
and T-shirts
in this week's
free-to-enter
competition.**

them. Circle the differences on carton B and seal the cartoon and coupon in an envelope. Write clearly the number of differences you found on the back of the envelope.

Post your entry to Intrigue Software Competition, Home Computing Weekly, No. 1, Golden Square, London W1R 3AB. Entries close at first post on Friday August 24, 1984.

You may enter as many times as you wish, but each entry must be on an official coupon — not a copy — and sealed in a separate envelope.

Important: please follow carefully the guidelines on entering — incomplete coupons and entries in envelopes with no numbers on the back cannot be considered. If you are a winner, the coupon will be used as a label to send your prize so clear writing is essential.

The rules

Entries will not be accepted from employees of Argus Specialist Publications, Intrigue Software and Alabaster Passmore & Sons. This restriction also applies to employees' families and agents of the companies.

The How to Enter section forms part of the rules.

Intrigue Software Competition

Entry Coupon

Name _____

Address _____

_____ post code _____

Number of differences found _____

T-shirt size (small, medium or large) _____

Complete clearly and fully — if you are a prizewinner this will act as a label. Post to Intrigue Software Competition, Home Computing Weekly, No. 1 Golden Square, London W1R 3AB. Closing date: first post, Friday August 24, 1984. Don't forget to follow closely the advice in the How to Enter section, including writing the number of differences on the back of your envelope.



HERE TO STAY



TEXAS INSTRUMENTS

HOME COMPUTER TI-99/4A £89.00

Console: 9900 Family, 16-bit microprocessor, plus 256-byte scratchpad RAM.

Memory: Total combined memory capacity: 110KBytes. Internal ROM memory supplied: 26KBytes. External ROM memory: (Solid State Software command modules) up to 36KBytes each. **RAM memory supplied:** 16KBytes (Expandable to 48KBytes). **Keyboard:** 48 key QWERTY with control and function keys (user definable), full upper and lower case capability, alpha lock, auto repeat.

"Parco Electrics is the largest retailer of Texas Instruments Home Computer products in the U.K. As its Managing Director, I offer my personal guarantee on all items we sell. Complete support from Texas Instruments now enables us to offer virtually limitless stocks of peripherals and software.

Parco Electrics will be pleased to help you in any way we can. Write today, enclosing a s.a.e. for our free price list and further information".

Francis Parrish.

PERIPHERALS AVAILABLE:

32K expansion, RS232, Disc Drives, Control Cards, Speech Synthesiser, Joysticks, Cassette Leads

SOFTWARE FROM THE U.S.A.

Miner 20 49er, Buck Rogers, Startreck, Burger Time, Parsec, Pole Position, and 100s more.....

All in stock.

EXTENDED BASIC and MINI MEMORY also in stock.

PARCO
Electrics

For a free price list and further information send a s.a.e. to:

4 DORSET PLACE, NEW STREET, HONITON, DEVON,
EX14 8QS

TELEPHONE (0404) 44425

Spiders/ Simon/Granny Unexpanded Sord M5 £12.95

CGL, CGL Hse, Goldings Hill,
Loughton, Essex IG10 2RR

In **Spiders**, you must move a little figure to the other side of the screen. The difficulty is that you are pursued by large, venomous spiders and you must avoid their webs. The idea is good and the characters are well defined. The only problem is that it's incredibly difficult to get a decent score, so you're left feeling frustrated and angry!

Simon is the M5 version of the electronic game. The object is to correctly repeat an increasing sequence of colours, numbers and sounds. This is a good

implementation but I found the original electronic game boring and this version didn't interest me either. I suppose you could get hooked if you like the original though.

In **Granny**, you are in control of the green man whose task is to get the old ladies across the busy road. The storyline is good but the sound and graphics could have been infinitely better.

These games are fun for the first few goes but would probably pall after that. It's a shame machine code was not used.) S . S .

instructions	50%
playability	60%
graphics	40%
value for money	50%



Pengi TI-99/4A £5

Stainless, 10 Alstone Rd, Stock-
port, Cheshire SK4 5AH

Pengi is an original game in which a small character is being chased by yetis. (Judging by the scenery, the action takes place on a mountain range.)

On the first screen, you must kill two yetis, in the second, four, and in the third, six, and so on.

To kill a yeti, you must use the space bar to push a block of ice at him. This isn't as simple as it sounds, because each block of ice must travel over at least one space before hitting a yeti. If an ice block is pushed into another ice block before moving over any spaces, it will shatter.

Things get really tough when the screen becomes very yeti-infested. Every time a yeti moves into a block of ice, the ice vanishes, and the more yetis, the less ice.

Only a very small portion of the screen is used to display the action. This, as well as the lack of sound, is probably to speed the game up, but it does not make full use of the TI's facilities.

However, very good graphics are used in the top half of the screen to show beautiful, snow-capped mountains. J.J.

instructions	70%
playability	70%
graphics	85%
value for money	60%



Cassette File Handling TI-99/4A £3.50

B. Jackson, 21 Rowan Way,
New Balderton, Newark, Notts
NG24 3AU

An Extended BASIC cassette filing program for all those who have either been unable to master file processing or don't have the time or patience to produce their own routines. In some respects it is better than the Personal Record Keeping module, in others, worse.

For the price, I reckon it's very good value. Each file consists of up to 100 'cards', and each card can contain up to four items of information. The name of a file can be up to 14 characters, and the names of the items of information can be up to 24 characters. Each item of information can be up to 23 characters, allowing you to store a greater amount of data per item than the 15 offered by the PRK module.

However, only four items per page places some restriction on, for example, an address book file. Among the options are Insertion, Deletion, Addition of items, and alphanumeric sorting by item. Documentation is sparse, but then little is needed for a program which is largely self-explanatory. For large files, however, cassette storage is impractical and rather risky, due mainly to the problem of shedding oxide and consequent drop-out. P.B.

instructions	85%
ease of use	95%
display	85%
value for money	100%



Fun, and a bit more serious

Play the games... then try the
utilities. But read what our
reviewers have got to say first

Slot Expanded Sord M5 £5.95

CGL, CGL House, Goldings
Hill, Loughton, Essex IG10 2RR

Slot is an implementation of the popular gambling game, One-armed Bandit. When it's loaded, the reels are drawn and you start with a fixed amount of credit. When you press 1, the reels change — absolutely no animation — and your credit is adjusted accordingly to the result of the reels.

You may be given the chance to Nudge or Hold as in the real version. I'm a terrible player and have sworn never to play this game, but in this version, I won money without even looking at the reels and in the end I was desperately trying to run out of

money with no success! Games shouldn't be too difficult but this is ridiculous.

Technically, the program is just better than awful. What happened to sound effects and smooth animation?

I've always thought implementing gambling games on computers was a waste of time since the attraction of the real thing was the thought of making money and this program does nothing to change my view. Quite honestly, this program wouldn't even be worth publishing as a magazine listing. S.S.

instructions	50%
playability	0%
graphics	10%
value for money	0%



Music Maker/ TI Ludo/ Charset 1 TI-99/4A £5.95

PiKaDee, 35 Parker St, Preston,
Lancs PR2 2AH

Three programs, the most outstanding of which is **TI Ludo**. It's worth the money for this alone. The game is for one to four players. The computer will participate if requested but, I warn you, it does have an uncanny knack of winning.

Music Maker is a brave attempt at a difficult program to produce, for creating tunes on the TI. Those you compose may be saved in files and loaded when you want to play them again. The example is a rendering of *Lily the Pink*.

The screen display shows two staves. Using the cursor keys, notes, from F below Middle C to F flat three octaves above this, may be entered. Complete options are included to change mistakes and edit notes in memory. Tunes may be played as two or three note chords, or with a very effective echo.

Unless you have a basic knowledge of music, you'll find this difficult to understand and use.

Charset 1 contains character sets re-defined and their hexadecimal codes stored in DATA statements to use in your own programs. J.W.

instructions	80%
playability	70%
graphics	50%
value for money	90%



**Practise
makes
perfect**

Get the picture

Instead of hi-resolution graphical portrayals of the characters and objects, there are blobs seemingly

**Proud ZX81
owner**

Cheat for Revenge

Pricey pens?

Another point is that these pens are not sold individually, although one colour is used much more frequently than the other

Character count

Good

looking

If your program does not
start at line 1, then change

Tommie McCall, Cramlington, Northumberland

Screen

extended

dy Hopkins, Poole,
rset

Working it out

```

10 POKE 23613,87
20 CLS
30 PAPER 0: CLS: FOR
  q=1 to 300: NEXT q:
  PAPER 7: CLS
40 PRINT at 21, 0; "(c)
  1982 Sinclair Research
  Ltd."
50 if INKEY$ = "" THEN
  GOTO 50
60 CLS
70 PRINT at 21, 0;
  FLASH 1; "K"
80 if INKEY$ = "@"
  THEN GOTO 100
90 GOTO 80
100 PRINT at 21, 0;
  FLASH 1; "Entry
  Given"

```

I get your magazine every week and I think it's great.
**Philip Rawlins, Belhaven,
East Lothian**

Send your letters to Letters, Home Computing Weekly, No.1 Golden Square, London W1R 3AB. Don't forget to name your computer — the best letter could win £5-worth of software. Got a problem with your micro? We'll soon be starting a queries page, so send your technical questions to Queries at the above address.

COMPETITION

NOW is the time for all TI-99/4A users to get out their pens and spot some differences. The reward for the first 50 winners is a package of software from Warwick Software, of 40 Kingsway, Newby, Scarborough, N. Yorks.

Each package comprises three games (Hopbit, Caveman and Diver) together with the Typing Tutor — and each package is worth more than £21.

Hopbit is a variation on the Frogger-type game. Your task is to guide the frog across the road — avoiding cars and trucks along the way — until you reach the river. Then jump on the back of a turtle, and make your way across river — but watch out for hungry crocodiles.

Once across the river, you simply jump down a grate which gets you home.

The second game is Caveman, an underground exploration game.

Go down a ladder to reach the first level — and all along you have to beware of various obstacles.

A total of five levels have to be passed through before you get the chance to sit on the King's throne at the end of a tunnel which leads off from level five. Once you have achieved this, you're a winner.

We stay a long way down for the third game, but this time not below ground — below water.

Diver gives you the chance to show your swimming skills, and make money into the bargain.

You are in control of a diver whose job it is to recover differing items from the seabed. These include gold coins and human bones.

A ship passes above you, and you have to catch the ship and place

Four of the best from Warwick Software

This week's competition gives all TI-99/4A users the chance to win three great games, and a program to help you improve your typing skills

your bounty in it.

Problems come in the form of ravenous sharks and an octopus.

All three of the games are for Extended BASIC, and Caveman also requires a joystick.

If you want to give the games a rest for a few minutes, and would prefer to spend your time learning — then the fourth program is right up your street.

Typing Tutor, for TI BASIC, is exactly what it says. The program starts off by taking you through the fundamentals — like which key does what.

From that point it gives you training until you reach the final part of the program which is a speed trial.

You will agree that these are fantastic prizes, and all you have to do to stand a chance of winning one of these packages is to try and spot as many differences as possible in the two cartoons below. Good luck.

coupon in an envelope. Write clearly the number of differences you found on the back of the envelope.

Post your entry to Warwick Competition, Home Computing Weekly, No.1 Golden Square, London W1R 3AB. Entries close at first post on Friday September 14, 1984.

You may enter as many times as you wish, but each entry must be on an official coupon — not a copy — and sealed in a separate envelope.

Important: please follow carefully the guidelines on entering — incomplete coupons and entries in envelopes with no numbers on the back cannot be considered. If you are a winner, the coupon will be used as a label to send your prize so clear writing is essential.

The rules

Entries will not be accepted from employees of Argus Specialist Publications, Warwick, and Alabaster Passmore & Sons. This restriction also applies to employees' families and agents of the companies.

The How to Enter section forms part of the rules.

How to enter

Study the two cartoons — there are a number of differences between them. Circle the differences on cartoon B and seal the cartoon and

Warwick Competition

Entry Coupon

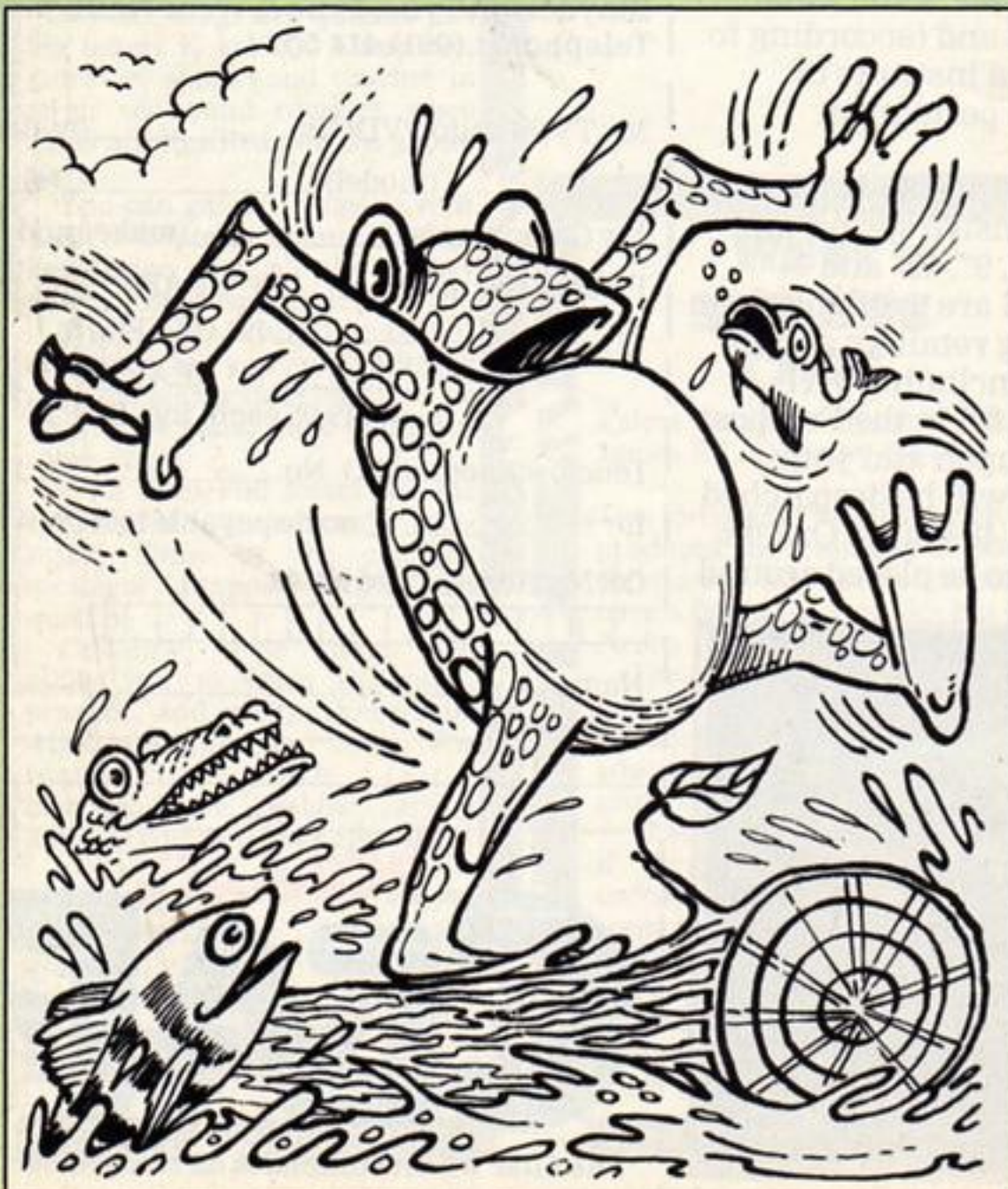
Name _____

Address _____

_____ post code _____

Number of differences found _____

Complete clearly and fully — if you are a prizewinner this will act as a label. Post to Warwick Competition, Home Computing Weekly, No.1 Golden Square, London W1R 3AB. Closing date: first post, Friday September 14, 1984. Don't forget to follow closely the advice in the How to Enter section, including writing the number of differences on the back of your envelope.



Our hero has done his gold prospecting and is now faced with the hazardous problem of transporting his treasure to the bank. Help him out in this program by Mike Willey

Take a break from violent games and give yourself a rest.

This game requires manoeuvrability. Krazy Kidd is our hero and what you have to do is move gold from his cave to the bank. He's finally realised that it pays to save — and also keeps itchy fingers away from his hoards of treasure.

However, poor Krazy is clumsy and not too co-ordinated. He has problems with walls — that is, he walks into them. He's also been known to fall down ladders. You must guide him and try to shield him from these pitfalls.

If a disaster occurs and he finds himself in a crumpled heap at the bottom of a ladder, or with a buckled nose from head-on contact with a concrete wall, you lose vital points.

Use the arrow keys to move. Watch out for the biggest hazard of all: dramatic subsidence in the cave, in which case you'll have to get the pickaxe to dig out all the rubble.

Variables

x, y character movement
sc score
P, Q cave collapse
d, t print position
T number of gold blocks
a\$ hold print messages

How it works

100-150 clear screen, define characters
160-260 set colours
270-300 set screen one title characters
310-390 set title
400-530 title graphics movement
550-590 subroutine for title
600 reset data line
620-860 set screen two characters
870-930 variables
940-960 initial position of man
970-1020 print score
1040-1120 random number generator, cave fall routine
1130-1160 detect key press
1180-1600 movement, collision detection
1610-1740 data
1750-1910 fall routine
1920-1990 change character
2000-2170 dig cave routine
2180-2220 pick up gold, reset position of pickaxe
2230-2400 gold in bank, message and score
2410-2500 trap in cave routine
2510-2600 collision routine, sound
2610-2710 end screen, final score

There's g the



old in m thar hills

```

100 CALL CLEAR
110 RANDOMIZE
120 FOR I=1 TO 15
130 READ Z,Z$
140 CALL CHAR(Z,Z$)
150 NEXT I
160 CALL COLOR(1,4,1)
170 CALL COLOR(2,13,1)
180 FOR I=3 TO 8
190 CALL COLOR(I,2,1)
200 NEXT I
210 CALL COLOR(9,2,1)
220 CALL COLOR(10,6,1)
230 CALL COLOR(12,7,1)
240 CALL COLOR(13,11,1)
250 CALL COLOR(14,10,4)
260 CALL COLOR(15,14,1)
270 CALL CLEAR
280 CALL HCHAR(15,3,144,28)
290 CALL VCHAR(1,4,40,12)
300 CALL VCHAR(1,28,40,12)
310 A$=" KRAZY KIDD"
320 T=5
330 GOSUB 550
340 A$=" *****"
350 T=7
360 GOSUB 550
370 A$="BY MICK WILLEY"
380 T=11
390 GOSUB 550
400 X=13
410 Y=3
420 CALL HCHAR(X,Y,107)
430 CALL HCHAR(X+1,Y,98)
440 FOR I=1 TO 13
450 Y=Y+2
460 CALL HCHAR(X,Y-2,32)
470 CALL HCHAR(X+1,Y-2,32)
480 CALL HCHAR(X,Y,107)
490 CALL HCHAR(X+1,Y,98)
500 CALL SOUND(40,-6,0)
510 NEXT I
520 FOR I=1 TO 200
530 NEXT I

```

```

540 GOTO 610
550 D=9
560 FOR I=1 TO LEN(A$)
570 CALL HCHAR(T,D+I,ASC(SEG$(A$,I,1)))
580 NEXT I
590 RETURN
600 RESTORE 1660
610 CALL CLEAR
620 FOR A=1 TO 22
630 READ G,H,J,K
640 CALL HCHAR(G,H,J,K)
650 NEXT A
660 FOR A=1 TO 23
670 READ G,H,J,K
680 CALL VCHAR(G,H,J,K)
690 NEXT A
700 A$=" SCORE"
710 P=2
720 D=1
730 GOSUB 830
740 A$="BANK"
750 P=1
760 D=14
770 GOSUB 830
780 A$="CAVE"
790 P=11
800 D=11
810 GOSUB 830
820 GOTO 870
830 FOR I=1 TO LEN(A$)
840 CALL HCHAR(P,D+I,ASC(SEG$(A$,I,1)))
850 NEXT I
860 RETURN
870 P=6
880 Q=26
890 SC=50
900 CALL HCHAR(P,Q,101)
910 X=11
920 Y=23
930 L=4
940 CALL HCHAR(X,Y,105)
950 CALL HCHAR(X+1,Y,96)

```



```

960 CALL GCHAR(X+2,Y,S)
970 IF S=32 THEN 1750
980 S$=" "&STR$(S)&" "
990 D=7
1000 FOR I=1 TO LEN(S$)
1010 CALL HCHAR(2,D+I,ASC(SEG$(S$,I,1)))
1020 NEXT I
1030 CALL KEY(0,K,S)
1040 I=INT(RND*250)+1
1050 IF I<>45 THEN 1130
1060 CALL GCHAR(13,13,S)
1070 IF S=100 THEN 1130
1080 FOR D=0 TO 25 STEP 5
1090 CALL SOUND(120,-6,D)
1100 CALL HCHAR(13,12,100,4)
1110 CALL HCHAR(14,12,100,4)
1120 NEXT D
1130 IF K=69 THEN 1180
1140 IF K=68 THEN 1260
1150 IF K=83 THEN 1390
1160 IF K=88 THEN 1530
1170 GOTO 940
1180 CALL GCHAR(X-1,Y,S)
1190 IF S<>32 THEN 2510
1200 X=X-1
1210 CALL HCHAR(X+2,Y,32)
1220 CALL SOUND(40,330,1)
1230 CALL HCHAR(X,Y,106)
1240 CALL HCHAR(X+1,Y,99)
1250 GOTO 1030
1260 CALL GCHAR(X,Y+1,S)
1270 IF S=100 THEN 2610
1280 IF S<>32 THEN 2510
1290 Y=Y+1
1300 CALL HCHAR(X,Y,107)
1310 CALL HCHAR(X+1,Y,98)
1320 CALL SOUND(50,-2,1)
1330 CALL HCHAR(X,Y-1,32)
1340 CALL HCHAR(X+1,Y-1,32)
1350 IF P<>X+1 THEN 1380
1360 IF Q=Y THEN 1370 ELSE 1380
1370 GOSUB 1920
1380 GOTO 1030
1390 CALL GCHAR(X+1,Y-1,S)
1400 IF S=129 THEN 2180
1410 IF S=103 THEN 2230
1420 IF S<>32 THEN 2510
1430 Y=Y-1
1440 CALL HCHAR(X,Y,107)
1450 CALL SOUND(40,-1,1)
1460 CALL HCHAR(X+1,Y,97)
1470 CALL HCHAR(X,Y+1,32)
1480 CALL HCHAR(X+1,Y+1,32)
1490 IF T<>X THEN 1520
1500 IF U=Y THEN 1510 ELSE 1520
1510 GOSUB 2000

```

```

1520 GOTO 1030
1530 CALL GCHAR(X+2,Y,S)
1540 IF S<>32 THEN 2510
1550 X=X+1
1560 CALL HCHAR(X-1,Y,32)
1570 CALL HCHAR(X,Y,106)
1580 CALL SOUND(50,220,1)
1590 CALL HCHAR(X+1,Y,99)
1600 GOTO 1030
1610 DATA 121,FFFFFFFFFFFFFFFFFFFF,1
44,FF222222FF888888,129,00000000
00707070,103,FFFFFFFFFFFFFFFFFFFF
1620 DATA 105,1038107CBABABABA,1
06,1039123C381C1A19
1630 DATA 96,3828282828282828,97
,384888844422202,98,382422424488
8880,99,3F21110904020101
1640 DATA 40,427E4242427E4242,10
1,0001424428100803,100,BFFBF7FFF
EDF7FFD
1650 DATA 104,50B892FF989C5A39,1
07,103810383A7C3838
1660 DATA 1,1,121,19,1,24,121,9,
24,1,121,10,24,26,121,7,21,10,12
1,17,7,2,121,12,14,5,129,4
1670 DATA 4,14,144,16,7,14,144,6
,7,22,144,8,10,4,144,13,10,25,14
4,5,13,22,144,5,13,29,144,3
1680 DATA 15,5,100,15,16,5,100,1
5,17,5,100,15,18,4,144,16,21,4,1
44,4,11,5,100,12,12,10,100,7
1690 DATA 18,22,144,8,2,14,103,2
,2,15,103,2
1700 DATA 2,1,121,22,21,10,121,3
,21,26,121,3,2,13,121,5,2,32,121
,22,10,4,144,12,10,22,144,4,8,19
,144,7
1710 DATA 7,22,144,4,14,22,144,5
,13,26,144,5,5,29,144,5,18,29,14
4,4,2,31,40,11,10,3,40,12
1720 DATA 13,28,40,3,14,31,40,10
,21,9,40,3,19,28,40,3,7,21,40,12
,8,18,40,7
1730 DATA 106,50B892FF989C5A39,1
05,103C167DBBF83838,107,103C167D
BBF83838
1740 DATA 108,344EA43C3C2C0C0C,1
09,085C889CFC8C8C4C
1750 I=2200
1760 CALL SOUND(40,I,0)
1770 X=X+1
1780 CALL HCHAR(X-1,Y,32)
1790 CALL HCHAR(X,Y,105)
1800 CALL HCHAR(X+1,Y,97)
1810 CALL GCHAR(X+2,Y,S)
1820 IF S<>32 THEN 1850
1830 I=I-55

```



```

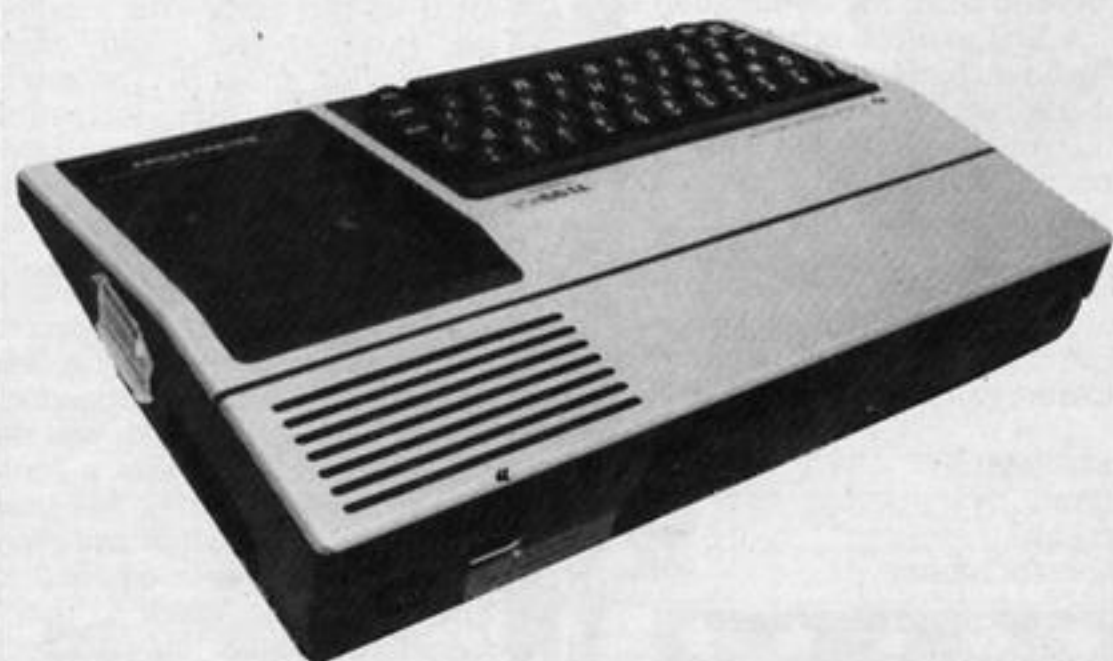
1840 GOTO 1760
1850 A$="HEY WATCH IT"
1860 FOR I=1 TO LEN(A$)
1870 D=11
1880 CALL HCHAR(23,D+I,ASC(SEG$(
A$,I,1)))
1890 NEXT I
1900 SC=SC-10
1910 GOTO 980
1920 RESTORE 1730
1930 FOR I=1 TO 3
1940 READ Z,Z$
1950 CALL CHAR(Z,Z$)
1960 NEXT I
1970 T=13
1980 U=16
1990 RETURN
2000 RESTORE 1740
2010 FOR A=1 TO 2
2020 READ Z,Z$
2030 CALL CHAR(Z,Z$)
2040 NEXT A
2050 FOR S=1 TO 4
2060 FOR I=1 TO 4
2070 CALL HCHAR(X,Y,108)
2080 FOR W=1 TO 50
2090 NEXT W
2095 CALL SOUND(50,-6,0)
2100 CALL HCHAR(X,Y,109)
2110 CALL HCHAR(X+1,Y,96)
2120 NEXT I
2130 CALL HCHAR(X,Y,32)
2140 CALL HCHAR(X+1,Y,32)
2150 Y=Y-1
2160 NEXT S
2170 RETURN
2180 CALL CHAR(105,"103C107FBF7F
3F3A")
2190 CALL HCHAR(X,Y,105)
2200 CALL HCHAR(X+1,Y-1,32)
2210 CALL HCHAR(P,Q,101)
2220 GOTO 1030
2230 RESTORE 1620
2240 FOR A=1 TO 5
2250 READ Z,Z$
2260 CALL CHAR(Z,Z$)
2270 NEXT A
2280 SC=SC+100
2290 L=L-1
2300 IF L=0 THEN 2410
2310 D=11
2320 CALL HCHAR(23,11,32,15)
2330 A$=" FANTASTIC!!"
2340 FOR I=1 TO LEN(A$)
2350 CALL HCHAR(23,D+I,ASC(SEG$(
A$,I,1)))
2360 NEXT I

```

```

2370 FOR I=110 TO 1100 STEP 110
2380 CALL SOUND(-200,I,0)
2390 NEXT I
2400 GOTO 980
2410 A$=" GAME OVER ."
2420 D=12
2430 FOR I=1 TO LEN(A$)
2440 CALL HCHAR(23,D+I,ASC(SEG$(
A$,I,1)))
2450 NEXT I
2460 FOR I=1600 TO 110 STEP -110
2470 CALL SOUND(-200,I,0)
2480 NEXT I
2490 CALL CLEAR
2500 GOTO 2630
2510 CALL SOUND(100,440,0)
2520 CALL SOUND(150,220,1)
2530 CALL HCHAR(23,11,32,13)
2540 D=12
2550 A$=" OUCH"
2560 FOR I=1 TO LEN(A$)
2570 CALL HCHAR(23,D+I,ASC(SEG$(
A$,I,1)))
2580 NEXT I
2590 SC=SC-5
2600 GOTO 1030
2610 CALL CLEAR
2620 PRINT " SORRY"::::
" YOU LOST HIM":::
2630 PRINT " YOU SCORED";S
C
2640 PRINT " "::
2650 PRINT " DO YOU WANT TO PLAY
AGAIN?":::
2660 PRINT " Y OR N"
2670 CALL KEY(D,K,S)
2680 IF K=89 THEN 600
2690 IF K=78 THEN 2700 ELSE 2670
2700 CALL CLEAR
2710 END

```



Keeping track of time

Do you need to know what date falls on which day? This program from R. W. Hatfield will help

If you have problems with dates, and we're talking about the calendar not the dried fruit, then this perpetual calendar program will be of invaluable help.

The screen displays the complete calendar for the year and month selected, and use of left and right cursor keys causes the following or preceding month to be scrolled on to the screen.

Conversion should be straightforward as no user-

How it works

100-200 instructions and title
210-420 initialisation
430-480 accepts input for year and month and converts month to numeric value
490-860 main loop as follows
520-580 evaluates instructions from keyboard
620-630 makes adjustments for ordinary and centennial leap years
670-760 calculates which day the month begins
810-910 provides screen display and calls for next instruction

defined graphics are involved.

Output to screen is entirely performed by display statements.

If you're the type of person that doesn't know whether it's Monday, Thursday or Christmas — this is the program for you.

Variables

Arrays: CALS, Month\$, Day\$
 hold data for screen display
Y input year
MDS input month
D month number
L leap year
T no. of complete years
S total of day advances for each year, leap year, and centennial leap year (every 400 years)
ML complete months
PL print list for dates
DS start point for day display

```
100 ! PERPETUAL CALENDAR
110 !
120 ! R.W.HADFIELD
130 !
140 CALL CLEAR
150 PRINT "      PERPETUAL CALENDAR
R      -----":
: "BASED ON THE GREGORIAN      CA
LENDAR ADOPTED BY GREAT      BRITAI
N IN 1752"
160 PRINT "-----
-----":
170 PRINT "ENTER YEAR & MONTH RE
QUIRED." : : "TO CHANGE DISPLAY
USE KEYS-" : : "S..FOR PRECEDING
MONTH" : : "D..FOR FOLLOWING MONT
H" : :
180 PRINT "ENTER..FOR NEW YEAR &
MONTH" : : "E..TO EXIT" : : "ANY KE
Y..TO CONTINUE"
190 CALL KEY(O,K,STA):: IF STA=0
THEN 190
200 CALL CLEAR :: PRINT "BY R.W.
HADFIELD      (C)1982" : : : : "DEP
RESS ALPHA LOCK" : : : :
210 !
220 ! INITIALISE
230 !
```

```
240 OPTION BASE 1
250 DIM CAL$(35)
260 DIM MONTH$(12)
270 DIM DAY$(13)
280 FOR A=1 TO 35
290 READ DATE$
300 CAL$(A)=DATE$
310 NEXT A
320 DATA 1,2,3,4,5,6,7,8,9,10,11
,12,13,14,15,16,17,18,19,20,21,2
2,23,24,25,26,27,28,29,30,31,,,
330 FOR B=1 TO 12
340 READ M$
350 MONTH$(B)=M$
360 NEXT B
370 DATA JANUARY,FEBRUARY,MARCH,
APRIL,MAY,JUNE,JULY,AUGUST,SEPT
EMBER,OCTOBER,NOVEMBER,DECEMBER
380 FOR C=1 TO 13
390 READ D$
400 DAY$(C)=D$
410 NEXT C
420 DATA MON,TUE,WED,THU,FRI,SAT
,SUN,MON,TUE,WED,THU,FRI,SAT
430 !
440 ! INPUTS
450 !
```


TI-99/4A PROGRAMS

```

460 CALL CLEAR :: DISPLAY AT(12,
3): "WHICH YEAR?" :: ACCEPT AT(12
,17)VALIDATE(DIGIT)SIZE(4)BEEP:Y
470 DISPLAY AT(14,3): "WHICH MONT
H?" :: ACCEPT AT(14,17)VALIDATE(
UALPHA)SIZE(9)BEEP:MD$
480 D=(POS("JANFEBMARAPR MAYJUNJU
LAUGSEP OCTNOVDEC",SEG$(MD$,1,3),
1)+2)/3 :: IF D=0 THEN 470 :: CA
LL CLEAR :: GOTO 620
490 !
500 ! START OF LOOP
510 !
520 IF CHR$(K)="E" THEN 910 :: I
F K=13 THEN 460 :: IF CHR$(K)="S
" THEN 530 :: IF CHR$(K)="D" THE
N 540 ELSE 900
530 D=D-1 :: GOTO 550
540 D=D+1
550 IF D=13 THEN D=1 ELSE 570
560 Y=Y+1
570 IF D=0 THEN D=12 ELSE 620
580 Y=Y-1
590 !
600 ! LEAP YEARS
610 !
620 L=(Y/4-INT(Y/4))>0
630 IF Y/100=INT(Y/100)AND NOT Y
/400=INT(Y/400)THEN L=-1
640 !
650 ! SET DAYS
660 !

```

```

670 T=Y-1
680 S=T+INT(T/4)-INT(T/100)+INT(
T/400)
690 RESTORE 740
700 FOR A1=1 TO D
710 READ ML :: S=S+ML
720 NEXT A1
730 READ PL :: IF L=0 AND D=2 TH
EN PL=PL+1
740 DATA 0,31,28,31,30,31,30,31,
31,30,31,30,31
750 IF L=0 AND D>2 THEN S=S+1
760 DS=S-(INT(S/7)*7)+1
770 !
780 ! DISPLAY
790 !
800 CALL CLEAR
810 DISPLAY AT(1,2):MONTH$(D):Y
820 FOR P=6 TO 24 STEP 3
830 DISPLAY AT(P,2):DAY$(DS):: D
S=DS+1
840 NEXT P
850 X=1
860 FOR CL=8 TO 24 STEP 4 :: FOR
R=6 TO 24 STEP 3
870 IF X>PL THEN 890
880 DISPLAY AT(R,CL):USING ("##"
):CAL$(X):: X=X+1
890 NEXT R :: NEXT CL
900 CALL KEY(3,K,STA):: IF STA=0
THEN 900 ELSE 520
910 END

```

AMSTRAD AND SPECTRUM

MICRO IQ 1000 MULTICHOICE
QUESTIONS/ANSWERS

RETURN OF THE JEDI —
SPECTRUM

EXCELLENT ARCADE GAME
WITH 8 SPEED LEVELS
BEAT THE HIGH SCORE AND WIN
A CASH PRIZE

AVAILABLE SHORTLY:

DRAWING PROGRAM FOR THE
AMSTRAD 64

ALL £5.50 INCL P&P. CHEQUES OR
POS ONLY TO:

M. K. CIRCUITS
116 SOUTH PARADE
GRIMSBY

TEXAS OWNERS GET INTO PRINT

At last, a 40 column thermal printer that just
plugs into the right hand port on your TI-99/4A.

List from Basic, ExBas etc. Works with any
module that asks for printer. e.g. PRK,
Household budget etc.

£145.00 inc VAT & carriage

Order to Arcade Hardware, 211, Horton Road
Fallowfield, Manchester, M14 7QE
Access orders to 061 225 2248

Wide range of modules also available.

If an advertisement
is wrong we're here
to put it right.

If you see an advertisement in the press, in print,
on posters or in the cinema which you find
unacceptable, write to us at the address below.

The Advertising Standards Authority. ✓

ASA Ltd, Dept 3 Brook House, Torrington Place, London WC1E 7HN

Meteor

**Dodge the meteors to dock the ship in this fast-moving game
written by Martin Gohrey**

As commander of a space vessel in the depths of a peril-filled galaxy you must attempt to dock your craft with the mothership.

Sounds easy? It would be if it wasn't for the meteors. They come thick and fast and it requires all your experience of commanding spaceships to achieve a successful docking whilst dodging the on-coming meteors.

If you're really skilled and manage to achieve the high score then you also have the chance to enter your name. Use S for right and O for left.

So have a go and look out for the meteors.

Variables

A,B,D meteor speeds
SC,HC score and high score

How it works

110-150 set variables
160-200 initialise characters
210-270 print title screen
370-540 set up game screen
550 starts game
560 displays scores
570-600 check for key and coincidence
610-640 move left routine
650-680 move right routine
690-710 coincidence check
720-730 docking check
740-770 game over routine
780-820 reset variables
830-840 restart games
860-920 successful docking and speed up

```
100 CALL CLEAR
110 HC=0
120 SC=0
130 D=18
140 B=15
150 A=10
160 CALL CHAR(99,"3C7ED57E3C")
170 CALL CHAR(42,"000814D3200000
00")
180 CALL CHAR(97,"00043A418EB24C
30")
190 CALL CHAR(98,"FFBDE7FF5AFFA5
5A")
200 CALL CHAR(96,"02020505")
210 FOR D=1 TO 12
220 CALL MOTION(#1,0,0)
```

```
230 CALL COLOR(D,16,2)
240 NEXT D
250 CALL COLOR(2,13,1)
260 CALL CLEAR
270 CALL SCREEN(2)
280 DISPLAY AT(2,6)BEEP:"METEOR
ATTACK"
290 DISPLAY AT(4,12)BEEP:"BY"
300 DISPLAY AT(6,7)BEEP:"M.GOHRE
Y 1984"
310 DISPLAY AT(8,2)BEEP:"THE AIM
OF THE GAME IS"
320 DISPLAY AT(10,4)BEEP:"TO DOC
K YOUR SPACECRAFT"
330 DISPLAY AT(12,3)BEEP:"WITH T
HE MOTHERSHIP BUT"
```


attack

```

340 DISPLAY AT(14,4)BEEP:"AVOID
THE METEORS!!!"
350 DISPLAY AT(17,4)BEEP:"PRESS
ANY KEY TO BEGIN"
360 CALL KEY(O,K,S):: IF S=0 THE
N 360
370 CALL CLEAR
380 CALL SCREEN(2)
390 CALL HCHAR(23,1,42,32)
400 CALL SPRITE(#28,98,3,1,120,0
,1)
410 CALL SPRITE(#1,96,14,170,120
)
420 CALL SPRITE(#2,97,7,10,10,0,
-A)
430 CALL SPRITE(#3,97,5,20,50,0,
B)
440 CALL SPRITE(#4,97,15,40,100,
0,B)
450 CALL SPRITE(#5,97,7,60,1,0,-
A)
460 CALL SPRITE(#6,97,5,70,10,0,
A)
470 CALL SPRITE(#7,97,15,70,50,0
,A)
480 CALL SPRITE(#8,97,5,130,1,0,
D)
490 CALL SPRITE(#9,97,7,130,100,
0,D)
500 CALL SPRITE(#10,97,15,140,50
,0,-B)
510 CALL SPRITE(#11,97,7,150,80,
0,A)
520 CALL SPRITE(#12,97,5,160,30,
0,-D)
530 CALL SPRITE(#13,97,5,50,1,0,
D)
540 CALL SPRITE(#14,99,12,90,1,0
,28)
550 CALL KEY(O,K,S):: IF S=0 THE
N 550
560 DISPLAY AT(24,1):"SCORE ";SC
:: DISPLAY AT(24,12):"HI ";HC;"
-" :: DISPLAY AT(24,19):A$
570 CALL KEY(O,K,S):: CALL COINC
(ALL,C):: IF C=-1 THEN 720 :: CA
LL MOTION(#1,-7,0):: IF S=0 THEN

```

```

690
580 IF K=68 THEN 650
590 IF K=83 THEN 610
600 GOTO 570
610 CALL MOTION(#1,-7,-7)
620 CALL COINC(ALL,C)
630 IF C=-1 THEN 720
640 GOTO 570
650 CALL MOTION(#1,-7,7)
660 CALL COINC(ALL,C)
670 IF C=-1 THEN 720
680 GOTO 570
690 CALL COINC(ALL,C)
700 IF C=-1 THEN 720
710 GOTO 570
720 CALL COINC(#1,#28,15,C)
730 IF C=-1 THEN 860
740 CALL SOUND(1000,-5,0)
750 CALL DELSPRITE(#1)
760 DISPLAY AT(12,8)BEEP:"GAME
OVER"
770 DISPLAY AT(14,7)BEEP:"PRESS
ANY KEY"
780 D=18
790 B=15
800 A=10
810 IF SC>HC THEN 930
820 SC=0
830 CALL KEY(O,K,S):: IF S=0 THE
N 830
840 GOTO 360
850 GOTO 850
860 A=A+1
870 B=B+1
880 D=D+1
890 SC=SC+1
900 CALL SOUND(100,110,0)
910 CALL SPRITE(#1,96,14,170,120
,0,0)
920 GOTO 550
930 HC=SC
940 DISPLAY AT(13,17):"YOU RANK
No.1 ENTER YOUR
NAME"
950 ACCEPT AT(24,19)BEEP:A$
960 GOTO 820

```


SUPERCHARGE YOUR TI-99/4A THIS CHRISTMAS

With something special

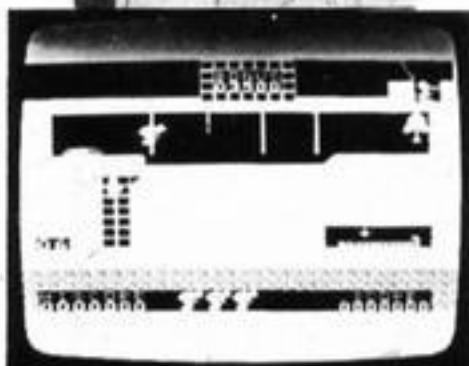
INTRIGUE PENTATHLON EXT. BASIC £5.95

FEATURES:

One or two players, Train in individual events or compete in the... INTRIGUE PENTATHLON. Two athletes competing together in the track events. Full colour graphics with incredibly realistic animation. Each event is packed full of REAL TIME ACTION that is hard to stop playing. Only the most competitive athletes will win.

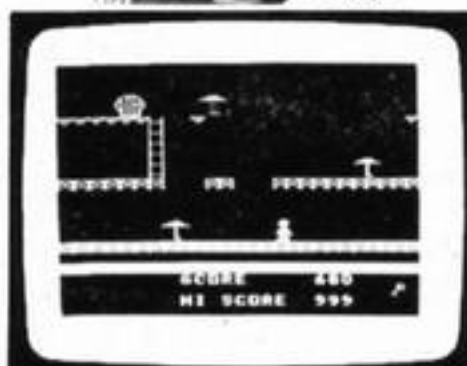
INTRIGUE PENTATHLON is five exciting games for the price of one.

Our Extended Basic range



Quasimodo help. 100% sprite graphics. 20 different scenes to clamour across. The action is fast and furious. Full sound. Keys or Joysticks.

A wonderful gift at £7.95

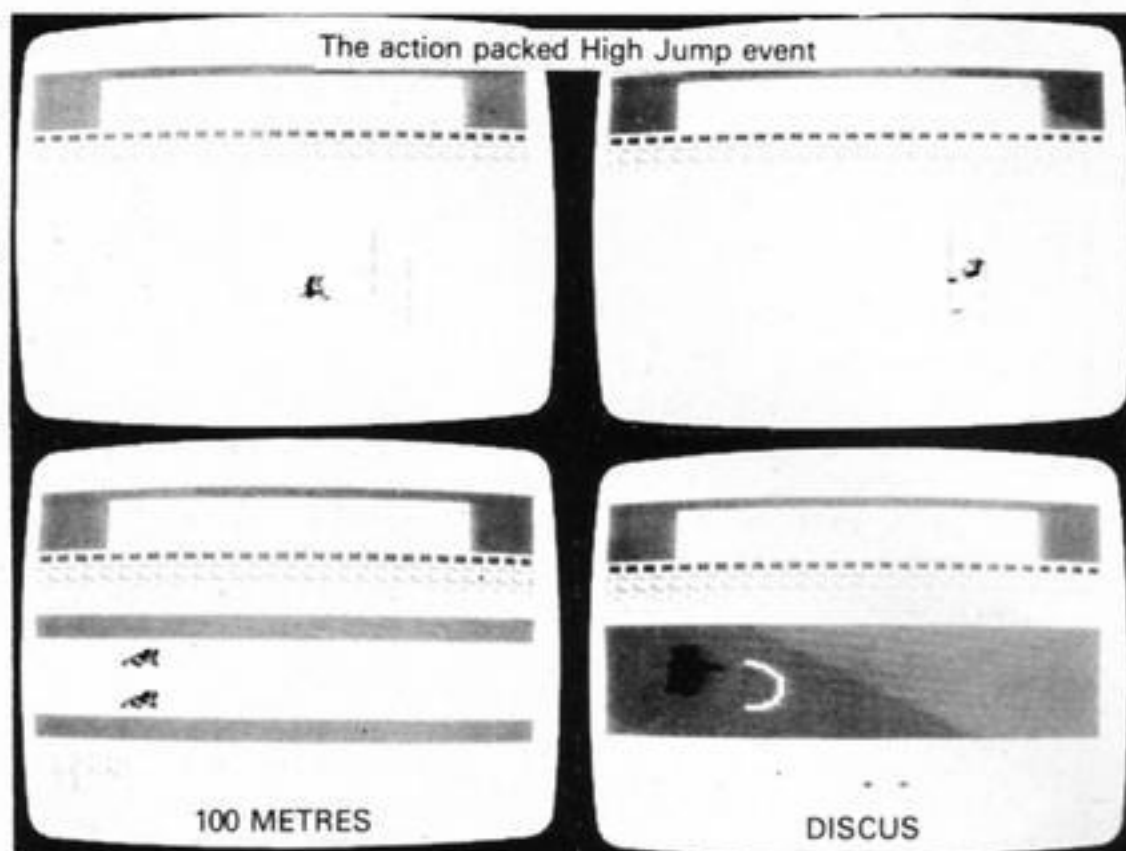


Lionel and the Ladders. This multi-screen game will more than amuse you. Just wait until Lionel meets his Princess.

Incredible value at £7.95



Snout of Spout. On the planet Spout the inhabitants are all called Snout. Can you walk a Snout and save the planet Spout. Only £7.95 to find out.



Be a sureshot with a Quickshot 1 that will plug straight into your TI in time for Christmas. Our Quickshot joystick will make someone very happy, the perfect surprise present.

Only £13.70 including Postage

Our Basic range (No Command module needed)



Santa and the Goblins. Our Christmas adventure is not just for children. Another mystery to solve.

All the family fun. 5.95



Adventuremania. This full graphic and text adventure is the only cassette game for TI to feature in the top 100 games of 1984.

Endless fun for £5.95



Mania. (Adventuremania part 2). Software Today and HCW gave 5 stars and said "I would certainly recommend Mania".

£5.95



Atlantis. The game that offers more in Basic than you ever imagined possible. 84 entirely different scenes of under water animated adventure.

The very best value £6.95



Beneath the Stars. 5 skill levels. Animated Spacecraft. Select Key option.

Action packed at only £5.95

Games to remember



from

INTRIGUE SOFTWARE

Cranbrook Road, Tenterden, Kent TN30 6UJ
Telephone 05806 4726

All Intrigue games are sold in wallets with full colour inlays and concise instructions. We mail order first class fast. Post Free. (Overseas orders + £1.00 per game Postage & Packing) Send your PO/Cheques to the address below or phone your ACCESS order now.



Telephone 05806 4726



With this program by Mike Poskitt you can keep all the records you need. All you require is your unexpanded TI and a cassette player

This is a simple record-keeping program requiring only the expanded TI-99/4A and a cassette player for storing the records

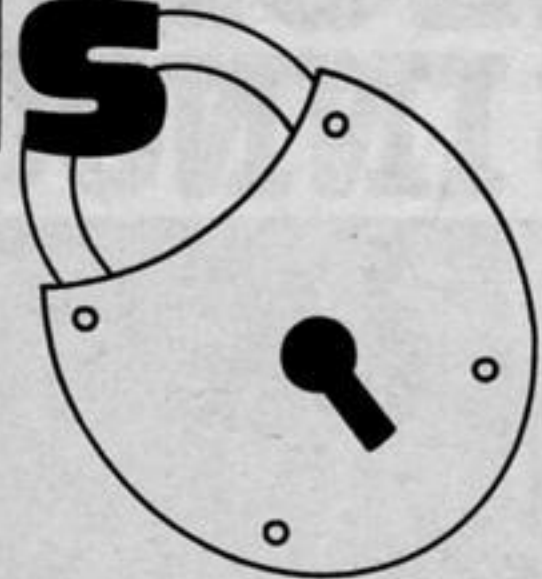
The program is menu-driven and the following selection is available:

- 1 Load file from cassette
- 2 Display file
- 3 Display file
- 4 Search file
- 5 Add a record
- 6 Modify a record
- 7 Save file on cassette
- 8 Exit

Hints on conversion

This program is quite straightforward and conversion should not present a problem. Peculiarities of TI BASIC

Save those records

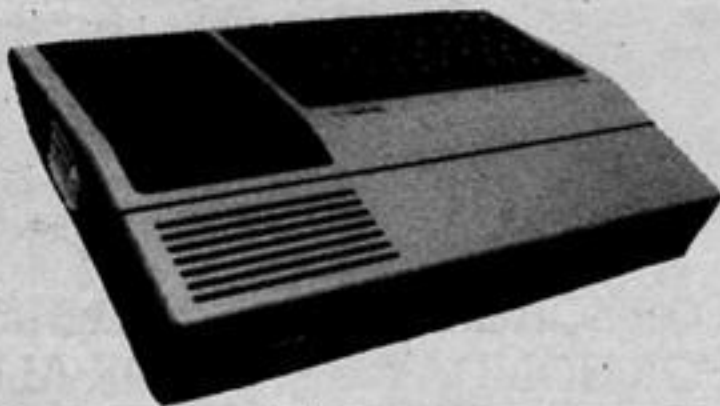


include single statement lines and built-in subroutines by using the CALL statement:

CALL CLEAR clears screen
CALL SCREEN sets screen colour, in this case yellow
CALL CHAR defines character, in this case a thin horizontal line, used to enhance the display
CALL COLOR (char set, foreground color, background color) sets colour of character, in this case a red line
CALL HCHAR (row, col, char code, no of repeats) place char on screen horizontal
CALL KEY (O,K,S;0 = key-unit, K = return variable, S = status variable) detects input from keyboard

Variables

Subscripted variable
RS(R) records
ISS(R) information
N number of records on file
RECS when searching for record, record name input from keyboard is stored in this variable



```
100 REM *****
110 REM * RECORD KEEPER *
120 REM *
130 REM * M.G.P. 6/84 *
140 REM *****
150 REM
160 CALL CLEAR
170 DIM RS(200), IS(200)
180 CALL CHAR(128, "0000FF")
190 CALL COLOR(13, 9, 1)
200 CALL SCREEN(12)
210 N=1
220 CALL CLEAR
230 REM *****MENU*****
240 PRINT TAB(8); "RECORD KEEPER"
250 PRINT "1. LOAD FILE FROM CAS
```

```
SETTE":
260 PRINT "2. CREATE NEW FILE":
270 PRINT "3. DISPLAY FILE":
280 PRINT "4. SEARCH FILE":
290 PRINT "5. ADD A RECORD":
300 PRINT "6. MODIFY A RECORD":
310 PRINT "7. SAVE FILE ON CASSE
TTE":
320 PRINT "8. EXIT":
330 CALL HCHAR(23, 1, 128, 32)
340 CALL HCHAR(5, 1, 128, 32)
350 PRINT "SELECT 1 TO 8"
360 CALL KEY(0, K, S)
370 IF (K<49)+(K>56) THEN 360
380 ON K-48 GOTO 400, 500, 710, 860
, 1020, 1160, 1380, 1460
390 REM ***LOAD FILE***
400 CALL CLEAR
410 N=1
420 OPEN #1: "CS1", INTERNAL, INPUT
, FIXED 128
430 INPUT #1: RS(N), IS(N)
440 IF RS(N)="END" THEN 470
450 N=N+1
460 GOTO 430
470 CLOSE #1
480 GOTO 220
490 REM *CREATE NEW FILE*
500 CALL CLEAR
510 PRINT "WARNING! YOU ARE CREA
TING A": "NEW FILE. IF A FILE ALR
EADY":
520 PRINT "EXISTS, IT WILL BE":
"ERASED.": "CONTINUE? (Y/N)"
530 CALL KEY(3, K, S)
540 IF S=0 THEN 530
550 IF K<>89 THEN 220
560 CALL CLEAR
570 N=1
```


PROGRAM

```

580 PRINT "ENTER RECORD AND INFO
RMATION":;"SEPARATELY.":;
590 PRINT "(TYPE ""END"" TO FINI
SH.):";
600 PRINT :;"RECORD":;
610 CALL HCHAR(23,3,128,6)
620 INPUT R$(N)
630 IF R$(N)<>"END" THEN 650
640 GOTO 220
650 PRINT :;"INFORMATION":;
660 CALL HCHAR(23,3,128,11)
670 INPUT I$(N)
680 N=N+1
690 GOTO 600
700 REM **DISPLAY FILE**
710 CALL CLEAR
720 IF N>1 THEN 750
730 PRINT "NO FILE IN MEMORY":;
:;:;:;:;
740 GOTO 820
750 FOR R=1 TO N
760 PRINT :;R$(R):;
770 IF R=N THEN 790
780 PRINT I$(R):;:;
790 INPUT "PRESS ENTER TO CONTIN
UE ":NUL$
800 CALL HCHAR(23,1,128,32)
810 NEXT R
820 FOR D=1 TO 600
830 NEXT D
840 GOTO 220
850 REM **SEARCH FILE**
860 CALL CLEAR
870 PRINT :;"FIND WHICH RECORD?"
:;:"(TYPE ""END"" TO FINISH)":;
880 INPUT REC$
890 IF REC$="END" THEN 220
900 FOR R=1 TO N
910 IF R$(R)=REC$ THEN 970
920 NEXT R
930 PRINT :;"RECORD NOT FOUND"
940 FOR D=1 TO 600
950 NEXT D
960 GOTO 860
970 CALL CLEAR
980 PRINT :;R$(R):;I$(R):;:;:;
990 INPUT "PRESS ENTER TO CONTIN
UE ":NUL$
1000 GOTO 860
1010 REM **ADD A RECORD**
1020 CALL CLEAR
1030 PRINT "ADDITIONS TO EXISTIN
G FILE.":;
1040 PRINT "(TYPE ""END"" TO FIN
ISH.):";
1050 PRINT :;"RECORD":;
1060 CALL HCHAR(23,3,128,6)

```

```

1070 INPUT R$(N)
1080 IF R$(N)<>"END" THEN 1100
1090 GOTO 220
1100 PRINT :;"INFORMATION":;
1110 CALL HCHAR(23,3,128,11)
1120 INPUT I$(N)
1130 N=N+1
1140 GOTO 1050
1150 REM *MODIFY A RECORD*
1160 CALL CLEAR
1170 PRINT "MODIFY WHICH RECORD?"
:;:"(TYPE ""END"" TO FINISH)":;
1180 INPUT REC$
1190 IF REC$="END" THEN 220
1200 FOR R=1 TO N
1210 IF R$(R)=REC$ THEN 1270
1220 NEXT R
1230 PRINT :;"RECORD NOT FOUND"
1240 FOR D=1 TO 600
1250 NEXT D
1260 GOTO 1160
1270 CALL CLEAR
1280 PRINT :;"RECORD: ";R$(R):;"
INFORMATION: ";I$(R):;:;
1290 PRINT "ENTER MODIFICATION..
":;
1300 PRINT "RECORD":;
1310 CALL HCHAR(23,3,128,6)
1320 INPUT R$(R)
1330 PRINT :;"INFORMATION":;
1340 CALL HCHAR(23,3,128,11)
1350 INPUT I$(R)
1360 GOTO 1160
1370 REM ***SAVE FILE***
1380 CALL CLEAR
1390 OPEN #1:"CS1",INTERNAL,OUTP
UT,FIXED 128
1400 FOR R=1 TO N
1410 PRINT #1:R$(R),I$(R)
1420 NEXT R
1430 CLOSE #1
1440 GOTO 220
1450 REM *****EXIT*****
1460 CALL CLEAR
1470 PRINT "WARNING! ANY FILE IN
MEMORY":;"WILL BE ERASED.":;:;:;
:;:;
1480 PRINT "CONTINUE? (Y/N)"
1490 CALL KEY(3,K,S)
1500 IF S=0 THEN 1490
1510 IF K<>89 THEN 220
1520 CALL CLEAR
1530 PRINT TAB(10);"GOODBYE!":;:;
:;:;:;
1540 FOR D=1 TO 600
1550 NEXT D
1560 END

```




Ten green bottles...

The odds are against you in this game by Tim Jackson. You must try to beat the computer at knocking bottles off walls

There are 21 bottles on a wall. You are an expert at throwing stones. Everytime you throw a stone you hit a bottle.

The computer is just as good as you. Computer and player alternate turns. They may take one, two or three stones each turn. The one left with the final bottle loses the game.

You may like to know you can't win!

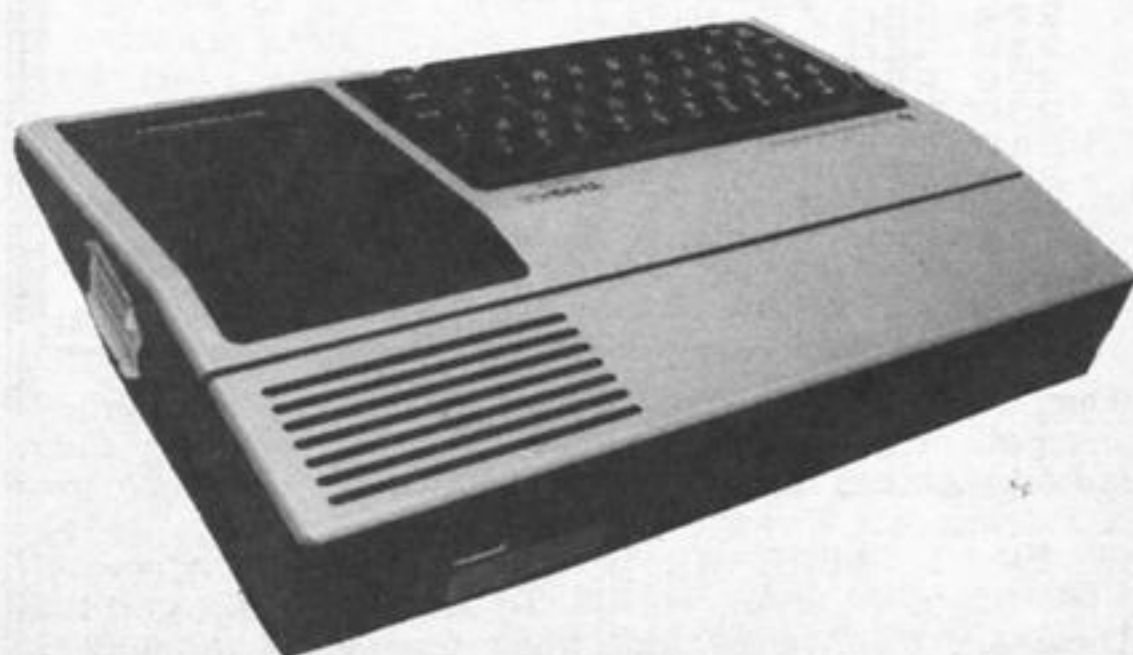
How it works

190-270 sets colours and characters
280-310 prints object of game
320-340 press space-bar routines
350-420 sets up game

430-610 asks how many stones player wishes to take
620-670 calculates number of stones computer takes
680-820 lose routine
830-870 print question routine
880-990 smash bottles routine

Variables

WL number of bottles
K key
S number of stones player takes
NS question or information
MS number of stones computer takes
A position of bottle to be smashed
X,D,N,I,G FOR...NEXT loops




```

100 REM *****
110 REM * BY TIM *
120 REM * JACKSON *
130 REM *****
140 REM *****
150 REM * BRICK A BOTTLE *
160 REM *****
170 CALL CLEAR
180 WL=21
190 CALL SCREEN(4)
200 CALL COLOR(9,7,2)
210 CALL COLOR(10,13,1)
220 CALL CHAR(100,"007F7F7F7F7F7F")
230 CALL CHAR(102,"00FEFEFEFEFEFEFE")
240 CALL CHAR(104,"1818183C7E7E7E7E")
250 CALL CHAR(105,"7E7E7E7E7E7E7E")
260 CALL CHAR(106,"8181C3C3CB8BB8BB")
270 CALL CHAR(107,"0024421824421842")
280 PRINT " BRICK A BOTTLE OR
THREE": " ~~~~~~
~~~": "THE GAME:-": " ~~~~~~
"
290 PRINT : "THERE ARE 21 BOTTLES
ON A WALL AND YOU ARE EXPERT
AT THROWING STONES, EVERY TIME"
300 PRINT "YOU THROW A STONE YOU
HIT A BOTTLE.": "I'M JUST AS GO
OD.": "WE MAY TAKE 1,2 OR 3 STON
ES EACH TURN."
310 PRINT "WE ALTERNATE TURNS": "
THE ONE WITH THE LAST BOTTLELOSE
S THE GAME": " (PRESS SPACE BAR
TO START)"
320 CALL KEY(0,K,S)
330 IF S=0 THEN 320
340 IF K<>32 THEN 320
350 CALL CLEAR
360 CALL SCREEN(8)
370 PRINT " HHHHHHHHHHHHHHHH
HHHHHHH": " I I I I I I I I I I I I
I I I I I"
380 FOR X=1 TO 5
390 PRINT "DFDFDFDFDFDFDFDFDFDFDF
DFDFDFDF": "DFDFDFDFDFDFDFDFDFDFDF
DFDFDFDF"
400 NEXT X
410 N$="HOW MANY DO YOU TAKE"
420 GOSUB 830
430 CALL KEY(0,K,ST)
440 IF ST=0 THEN 430
450 S=1
460 IF K<>49 THEN 490

```

```

470 GOSUB 880
480 GOTO 620
490 S=2
500 IF K<>50 THEN 530
510 GOSUB 880
520 GOTO 620
530 S=3
540 IF K<>51 THEN 570
550 GOSUB 880
560 GOTO 620
570 N$="TAKE 1,2 OR 3"
580 GOSUB 830
590 FOR D=1 TO 500
600 NEXT D
610 GOTO 410
620 MS=4-S
630 N$="I TAKE "&STR$(MS)
640 GOSUB 830
650 S=MS
660 GOSUB 880
670 IF WL>1 THEN 410
680 N$="1 BOTTLE LEFT I WIN !"
690 GOSUB 830
700 CALL SOUND(1000,131,0,165,5,
196,10,-3,28)
710 CALL SOUND(1000,262,0,330,0,
392,0)
720 CALL SOUND(1000,524,0,660,0,
784,0)
730 N$="WANT TO PLAY AGAIN?"
740 GOSUB 830
750 CALL KEY(0,K,ST)
760 IF ST=0 THEN 750
770 IF K=89 THEN 170 ELSE 780
780 N$="BETTER LUCK NEXT TIME"
790 GOSUB 830
800 FOR II=1 TO 500
810 NEXT II
820 END
830 CALL HCHAR(5,1,32,32)
840 FOR N=1 TO LEN(N$)
850 CALL HCHAR(5,4+N,ASC(SEG$(N$
,N,1)))
860 NEXT N
870 RETURN
880 A=31-WL
890 FOR I=1 TO S
900 CALL HCHAR(12,A,107)
910 CALL HCHAR(13,A,106)
920 FOR G=2 TO 30 STEP 2
930 CALL SOUND(100,-5,6)
940 NEXT G
950 CALL VCHAR(12,A,32,2)
960 A=A+1
970 NEXT I
980 WL=WL-S
990 RETURN

```


You'll need all your wits about you to avoid crashes. You're in control — or are you? in this game by Graham Baldwin

Have you ever wondered what it would be like to control the traffic at a busy crossroads? This simple game gives you the opportunity to find out, and will test your foresight and reactions to the limit.

Four cars are slowly approaching the crossroads and your job is to stop the traffic, one lane at a time, with the cursor keys to prevent collisions.

Periodically the cars will accelerate, making your task a little harder each time. If you allow a collision the other cars will stop briefly to enjoy the spectacle, then resume at their old speed, or slightly less if they were travelling at more than 20mph.

The game ends after 10 crashes, when the highest traffic speed and best speed so far are displayed.

You'll be doing well if you can keep the traffic running at 30mph but the cars have a theoretical maximum speed of 126 mph, which should keep even dedicated button-pushers on the hop.

Variables

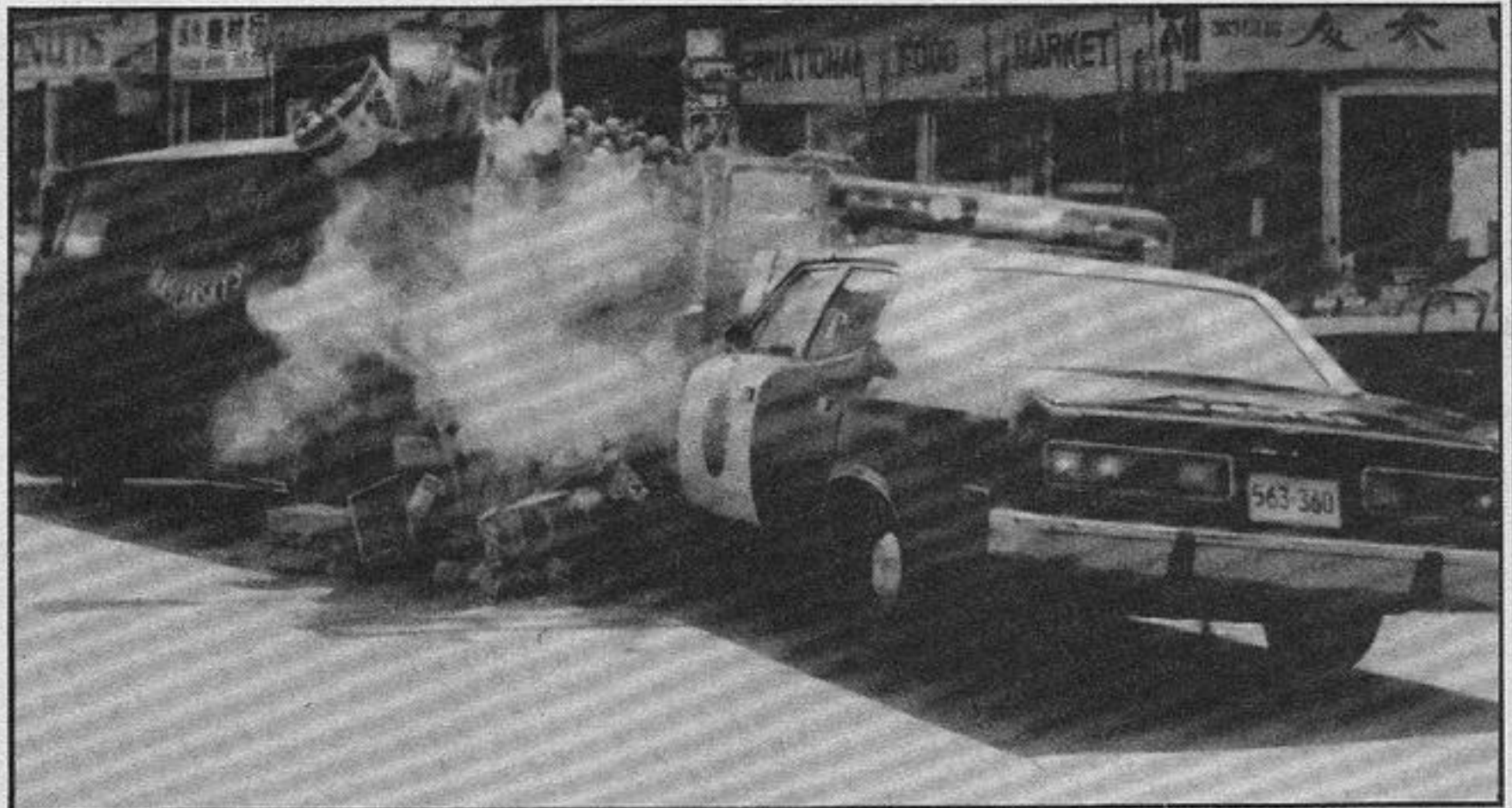
SP speed of cars
CR number of crashes
CT counter for increasing speed
BST best speed so far

How it works

140 clear screen
150 set sprite magnification
160-170 sub-program calls
190 set variables
200-210 sub-program calls
220-240 check for crash and number of crashes
250 scan keys
260 counter for speed increase
270-300 stop cars if key is pressed
330-410 end of game

```
100 REM      *TRAFFIC COP*
110 REM
120 REM GRAHAM BALDWIN 1984
130 REM
140 CALL CLEAR
150 CALL MAGNIFY(3)
160 CALL GRAPHICS
```

Traffic cop



420-560 define characters
570-600 set colours
630-770 instructions
780-1010 place graphics on screen
1020-1090 place cars on screen
1100-1220 explosion when cars crash
1230-1290 increase speed of cars

Hints on conversion

As this program uses sprite graphics it cannot be directly converted to other computers without these facilities, unless you wish to use machine code. If you want to try conversion here are some TI Extended BASIC statements for reference.

CALL SPRITE (sprite number, character code, colour code, start row, start column, row velocity, column velocity) places a sprite on the screen and starts it moving.

CALL MOTION (sprite number, row velocity, column velocity) sets or alters the motion of a sprite.

CALL DELSPRITE (sprite number or ALL) removes sprites from the screen

CALL SUBPROG(VARIABLE) transfers control to a user-written sub-program with optional variables transferred between the main and sub-programs

```
170 CALL INSTRUCTIONS
180 CALL CLEAR
190 SP=8 :: CR=0
200 CALL SETUP(SP)
210 CALL CARS(SP)
220 CALL CDINC(ALL,H):: IF H THE
N CALL CRASH(CR)ELSE 250
230 CALL SOUND(-100,1000,5)
240 IF CR>9 THEN 320 ELSE 210
250 CALL KEY(3,K,S)
260 CT=CT+1 :: IF CT>75 THEN CAL
L FASTER(CT,SP,BST)
270 IF K=83 THEN CALL MOTION(#1,
0,0)ELSE CALL MOTION(#1,0,SP)
280 IF K=68 THEN CALL MOTION(#2,
0,0)ELSE CALL MOTION(#2,0,-SP)
290 IF K=69 THEN CALL MOTION(#3,
0,0)ELSE CALL MOTION(#3,SP,0)
300 IF K=88 THEN CALL MOTION(#4,
0,0)ELSE CALL MOTION(#4,-SP,0)
310 GOTO 220
320 CALL CLEAR
330 DISPLAY AT(6,1):"YOU CAUSED
10 CRASHES, WITH"
340 DISPLAY AT(8,1):"THE TRAFFIC
RUNNING AT"
350 DISPLAY AT(10,1):SP;"MPH"
360 DISPLAY AT(12,2):"BEST SPEED
SO FAR IS";BST;"MPH"
370 DISPLAY AT(16,6)BEEP:"PLAY A
GAIN? (Y/N)"
380 CALL KEY(3,K,S):: IF S=0 THE
N 380
```


PROGRAM

```

390 IF K=78 THEN END
400 IF K<>89 THEN 370
410 CALL CLEAR :: GOTO 170
420 SUB GRAPHICS
430 CALL SCREEN(2)
440 CALL CHAR(128,"FF10101010101
OFF")
450 CALL CHAR(129,"818181FF81818
181")
460 CALL CHAR(130,"0101010101020
4F8")
470 CALL CHAR(131,"8080808080402
01F")
480 CALL CHAR(132,"1F20408080808
080")
490 CALL CHAR(133,"F804020101010
101")
500 CALL CHAR(136,"000000FFFF")
510 CALL CHAR(137,"1818181818181
818")
520 CALL CHAR(96,"000000007FFFFF3
F3F3F3FF7F0000000000000000FFFE1F
1F1F1FFEFF")
530 CALL CHAR(100,"00000000FF7FF
8F8F8F87FFF0000000000000000FEFFC
FCFCFCFFFE")
540 CALL CHAR(104,"0B0F0F0F0F0C0
C0C0F0F0C0C0F0F0F07D0F0F0F0F0303
030F0F03030F0F0F0E0")
550 CALL CHAR(108,"070F0F0F0C0C0
F0F0C0C0C0F0F0F0F0BE0F0F0F03030F
0F0303030F0F0F0F0D0")

```

```

560 CALL CHAR(112,"014163333F3F1
F1F1F1F3F7B63830301008183C6EEFCF
CF8F8F8FCFC9E0701")
570 CALL COLOR(13,16,2,14,16,1)
580 FOR I=2 TO 8
590 CALL COLOR(I,16,1)
600 NEXT I
610 SUBEND
620 SUB INSTRUCTIONS
630 DISPLAY AT(10,8):"TRAFFIC CO
P."
640 DISPLAY AT(13,2):"DO YOU WAN
T INSTRUCTIONS?"
650 DISPLAY AT(15,11)BEEP:"(Y/N)
"
660 CALL KEY(3,K,S):: IF S=0 THE
N 660
670 IF K=78 THEN CALL CLEAR :: G
OTO 750
680 IF K<>89 THEN 650
690 CALL CLEAR
700 DISPLAY AT(4,2):"USE THE ARR
OW KEYS (ESD,X)"
710 DISPLAY AT(6,3):"TO STOP THE
TRAFFIC AND"
720 DISPLAY AT(8,1):"PREVENT COL
LISIONS. WHEN TEN"
730 DISPLAY AT(10,3):"CRASHES HA
VE OCCURRED YOU"
740 DISPLAY AT(12,2):"WILL BE RE
MOVED FROM DUTY."
750 DISPLAY AT(18,4):"PRESS ANY

```



TI-99/4A PROGRAM

```

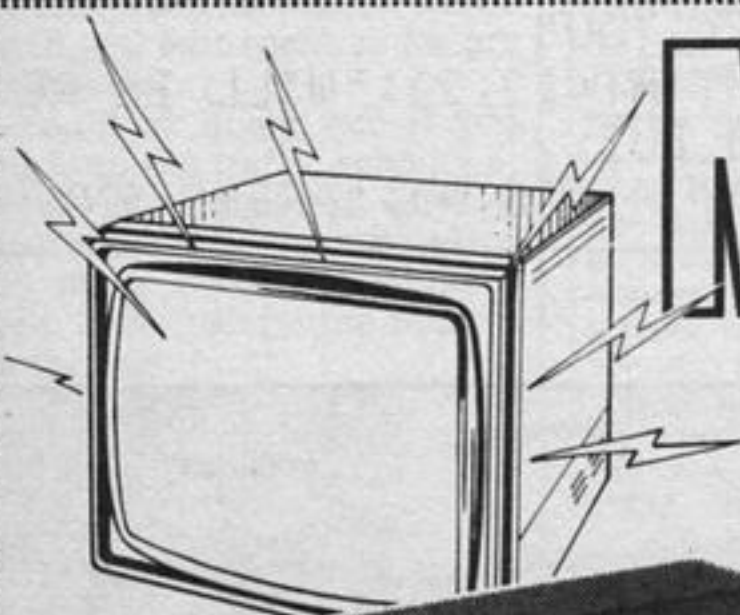
KEY TO PLAY."
760 CALL KEY(3,K,S):: IF S=0 THE
N 760
770 SUBEND
780 SUB SETUP(SP)
790 DISPLAY AT(4,1):"TRAFFIC"
800 DISPLAY AT(6,3):"CDP"
810 DISPLAY AT(4,21):"SPEED"
820 DISPLAY AT(6,20):SP;"MPH"
830 DISPLAY AT(20,1):"TI-99/4A"
840 DISPLAY AT(20,21):"CRASHES"
850 CALL HCHAR(9,1,128,32)
860 CALL HCHAR(15,1,128,32)
870 CALL VCHAR(1,13,129,24)
880 CALL VCHAR(1,19,129,24)
890 CALL VCHAR(1,14,32,120)
900 CALL HCHAR(10,1,32,160)
910 CALL HCHAR(9,13,130)
920 CALL HCHAR(9,19,131)
930 CALL HCHAR(15,13,133)
940 CALL HCHAR(15,19,132)
950 FOR I=2 TO 30 STEP 3
960 CALL HCHAR(12,I,136,2)
970 NEXT I
980 FOR I=1 TO 22 STEP 3
990 CALL VCHAR(I,16,137,2)
1000 NEXT I
1010 SUBEND
1020 SUB CARS(SP)
1030 IF SP>20 THEN SP=SP-4
1040 DISPLAY AT(6,20):SP;"MPH"
1050 CALL SPRITE(#1,96,14,75,1,0
,SP)

```

```

1060 CALL SPRITE(#2,100,16,98,25
0,0,-SP)
1070 CALL SPRITE(#3,108,6,1,128,
SP,0)
1080 CALL SPRITE(#4,104,3,190,10
5,-SP,0)
1090 SUBEND
1100 SUB CRASH(CR)
1110 CALL MOTION(#1,0,0,#2,0,0,#
3,0,0,#4,0,0)
1120 CALL SPRITE(#9,112,9,88,120
)
1130 CALL SPRITE(#10,112,12,88,1
20,5,5,#11,112,12,88,120,-5,-5,#
12,112,12,88,120,5,-5,#13,112,12
,88,120,-5,5)
1140 FOR I=0 TO 20
1150 CALL COLOR(#9,9)
1160 CALL SOUND(-200,-7,I,110,I+
5)
1170 CALL COLOR(#9,12)
1180 NEXT I
1190 CALL DELSPRITE(ALL)
1200 CR=CR+1
1210 DISPLAY AT(22,23):CR
1220 SUBEND
1230 SUB FASTER(CT,SP,BST)
1240 CALL SOUND(200,-2,0,500,5)
1250 SP=SP+2
1260 IF SP>BST THEN BST=SP
1270 CT=0
1280 DISPLAY AT(6,20):SP;"MPH"
1290 SUBEND

```



MEGA-SOUND

FOR SPECTRUM 48K

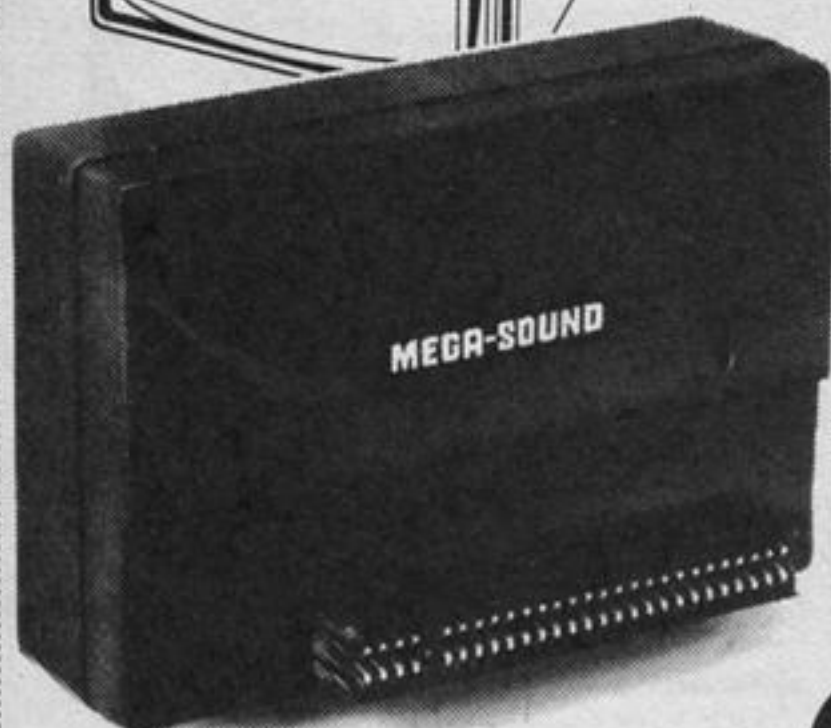
Compatible with ZX Spectrum +

**Achieve amazing sound capabilities
that your Spectrum has been lacking.**

The Cheetah Mega-Sound simply plugs into the user port at the rear of your computer and enables the Spectrum sound to be amplified through your T.V., providing you with a whole new sound dimension. The Spectrum "Beep" will also be amplified through the T.V., ensuring confidence when pressing keys during programming.

Truly Amazing

at £10.95



Prices include VAT, postage & packing.
Delivery normally 14 days.
Export orders at no extra cost.
Dealer enquiries welcome
Cheetah products available from
all good computer stores.

Cheetah
Marketing

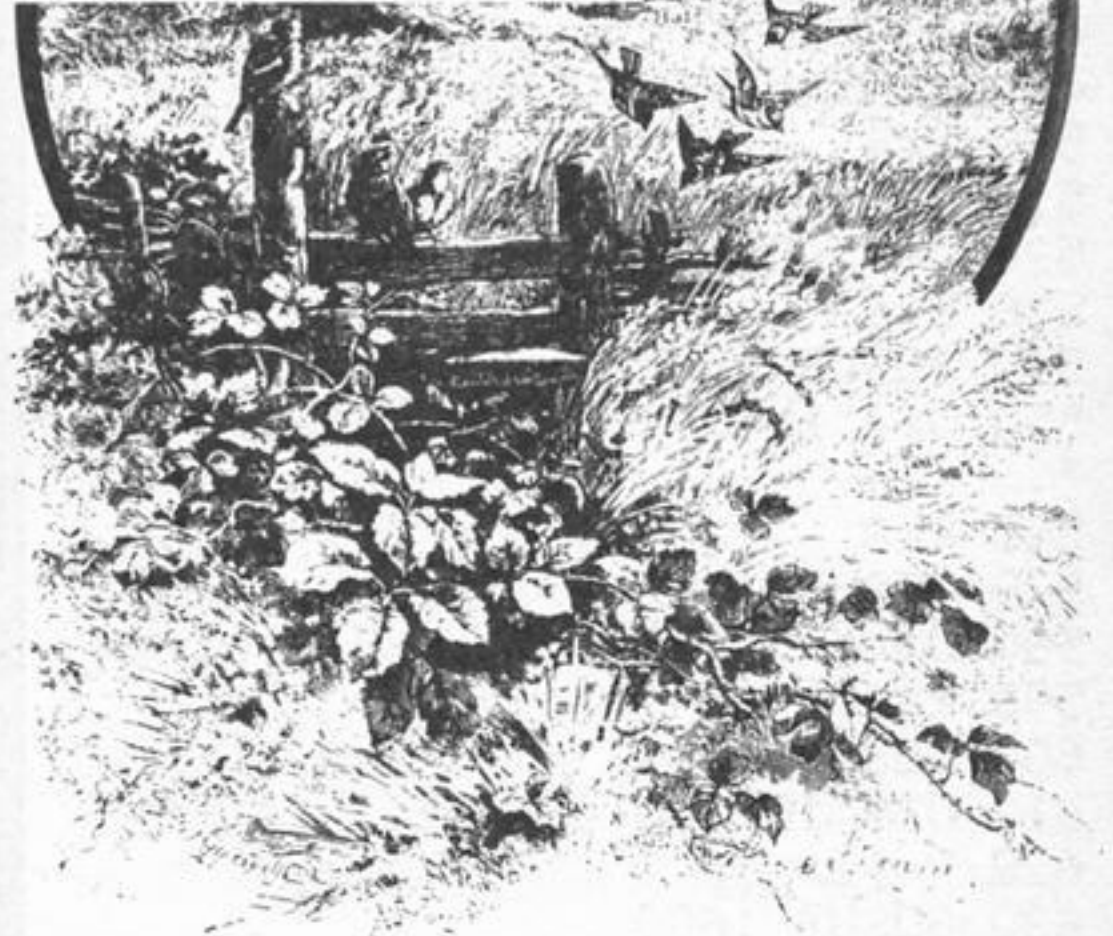


HCW
Cheetah Marketing Ltd, 24 Ray Street,
London EC1R 3DJ. Tel: 01 833 4909.
Telex: 8954958.

Pick the flowers in the garden — but watch out for deadly toadstools. You'll need swift reactions to be a success.
By Paul Twigg



Come into the garden, Maude



Are you fed up with TI games which pause for thought each time you press a key? This program was written to solve that problem.

All you have to do is pick the flower, avoiding the toadstools. Sounds simple? It's not. You never stop moving and you need nimble fingers to get a high score.

Keys E and D have been used so you can use one hand and be

How it works

100-190 set colour
200-250 define chars
260-430 title screen
440-830 instructions
840-970 set up screen
980-1160 main routine
1170-1300 pick flower
1310-1380 pick toadstool
1390-1450 print score
1460-1490 play again?
1500-1580 bonus scoring

1590!1630 print at routine
1640-1820 hi score, name routine

Conversions for other machines should be simple as this program is simple.

CALL CLEAR clears the screen
CALL HCHAR print at
CALL VCHAR print at
CALL GCHAR screen PEEK
CALL CHAR user defined graphics hex codes

Variables

SC score
W row number
B column number
N bonus score counter

quicker.

The amount of flowers remains constant but the toadstools increase in number as the games progress. They can crop up where you least expect them, so watch your step!

```
100 REM SET COLOURS
110 CALL CLEAR
120 CALL SCREEN(13)
130 CALL COLOR(1,8,15)
140 CALL COLOR(13,12,7)
150 CALL COLOR(14,8,1)
160 CALL COLOR(16,16,1)
170 FOR A=2 TO 12
180 CALL COLOR(A,14,15)
190 NEXT A
200 REM DEFINE GRAPHICS
210 CALL CHAR(32,"0000000000000000
FF")
220 CALL CHAR(130,"18306E7ADFFF1
818")
230 CALL CHAR(140,"10361C086B3E1
C")
240 CALL CHAR(150,"3844386CA384
```

```
406")
250 CALL CHAR(152,"18187E7E18181
818")
260 REM TITLE SCREEN
270 CALL HCHAR(1,1,140,32)
280 CALL HCHAR(24,1,140,32)
290 CALL VCHAR(1,1,140,24)
300 CALL VCHAR(1,32,140,24)
310 J$="THE FLOWER PICKER"
320 R=7
330 GOSUB 1600
340 J$="WRITTEN BY PAUL ROSS TWI
GG"
350 R=14
360 GOSUB 1600
370 J$="PRESS Y/N FOR INSTRUCTIO
NS "
380 R=21
```



```

390 GOSUB 1600
400 CALL KEY(0,K,S)
410 IF S=0 THEN 400
420 IF K=89 THEN 450
430 IF K=78 THEN 840 ELSE 400
440 REM INSTRUCTIONS
450 CALL CLEAR
460 J$=" THE FLOWER PICKER"
470 R=1
480 GOSUB 1600
490 J$="RUSH AROUND THE FIELD"
500 R=4
510 GOSUB 1600
520 J$="COLLECTING THE FLOWERS,"
530 R=6
540 GOSUB 1600
550 J$="BUT AVOID THE TOADSTOOLS"
560 R=8
570 GOSUB 1600
580 J$="AS THEY WILL KILL YOU."
590 R=10
600 GOSUB 1600
610 J$="IF YOU CAN PICK ENOUGH"
620 R=12
630 GOSUB 1600
640 J$="FLOWERS YOU WILL CHANGE"
650 R=14
660 GOSUB 1600
670 J$="COLOUR AND THEN YOU CAN"
680 R=16
690 GOSUB 1600
700 J$="PICK ONE TOADSTOOL."
710 R=18
720 GOSUB 1600
730 J$="USE KEYS E + D TO MOVE"
740 R=20
750 GOSUB 1600
760 J$="UP AND DOWN."
770 R=22
780 GOSUB 1600
790 J$="PRESS ANY KEY TO PLAY"
800 R=24
810 GOSUB 1600
820 CALL KEY(0,K,S)
830 IF S=0 THEN 820
840 CALL CLEAR
850 CALL CHAR(32,"")
860 FOR A=1 TO 12
870 CALL COLOR(A,2,1)
880 NEXT A
890 CALL SCREEN(13)
900 N=0
910 SC=0
920 RANDOMIZE
930 FOR Q=1 TO 30

```

```

940 CALL HCHAR(INT(RND*22+2),INT
(RND*29+3),140)
950 CALL HCHAR(INT(RND*21+3),INT
(RND*29+3),130)
960 NEXT Q
970 REM MAIN ROUTINE
980 REM MAIN ROUTINE
990 FOR W=1 TO 24
1000 FOR B=2 TO 32
1010 CALL GCHAR(W,B,X)
1020 IF X<>130 THEN 1040
1030 GOSUB 1320
1040 IF X<>140 THEN 1060
1050 GOSUB 1180
1060 CALL KEY(1,K,S)
1070 CALL HCHAR(W,B,150)
1080 CALL HCHAR(W,B,32)
1090 IF (K<>5)+(W<=1) THEN 1110
1100 W=W-1
1110 IF (K<>3)+(W>=24) THEN 1130
1120 W=W+1
1130 NEXT B
1140 IF W<>24 THEN 1160
1150 W=23
1160 NEXT W
1170 REM PICK FLOWER
1180 CALL SOUND(10,-5,0)
1190 N=N+1
1200 M=M+1
1210 SC=SC+10
1220 CALL HCHAR(INT(RND*22+2),IN
T(RND*29+3),140)
1230 IF N<>10 THEN 1250
1240 GOSUB 1510
1250 IF N<>15 THEN 1270 ELSE 126
0
1260 GOSUB 1550
1270 IF M<>5 THEN 1300
1280 CALL HCHAR(INT(RND*22+2),IN
T(RND*29+3),130)
1290 M=0
1300 RETURN
1310 REM PICK TOADSTOOL
1320 IF N>=10 THEN 1330 ELSE 134
0
1330 GOTO 1530
1340 FOR A=1 TO 30 STEP 2
1350 CALL HCHAR(W,B,152)
1360 CALL SOUND(100,-3,A)
1370 CALL SCREEN(RND*12+3)
1380 NEXT A
1390 REM PRINT SCORE
1400 CALL CLEAR
1410 IF SC>HS THEN 1420 ELSE 144
0
1420 HS=SC

```



```

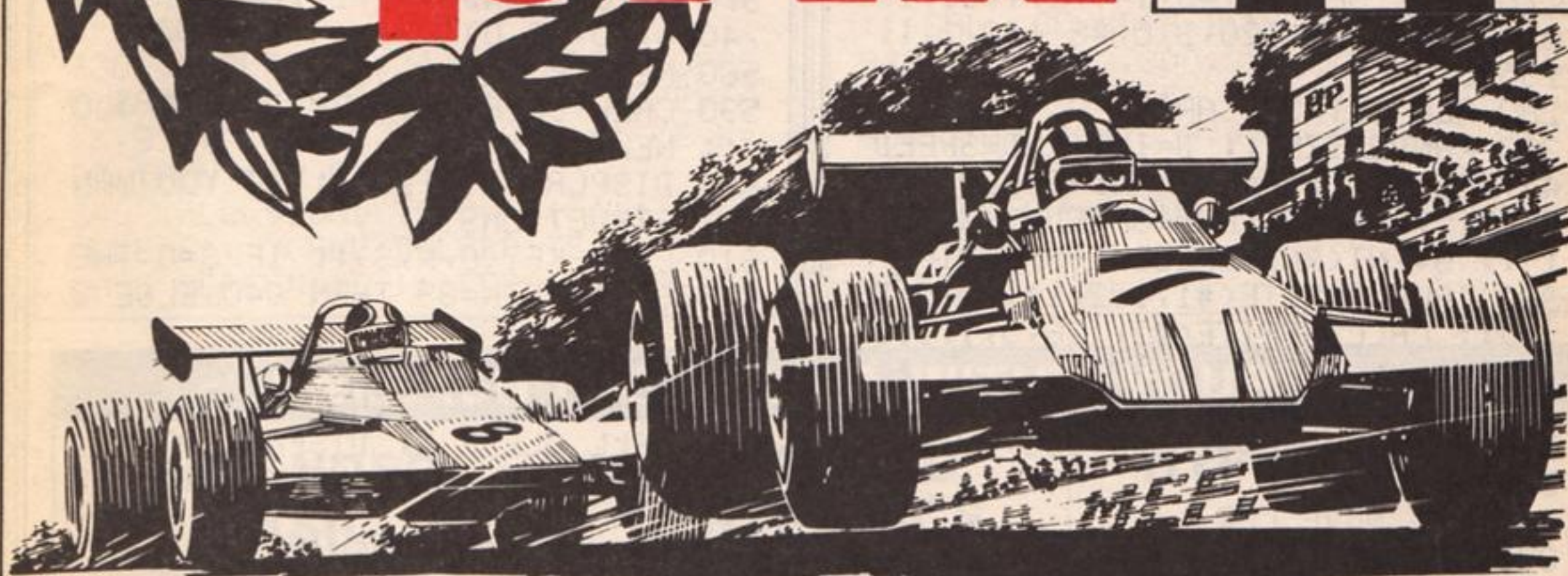
1630 RETURN
1640 REM HI SCORE
1650 J$="OH JOLLY GOOD! A NEW HI
SCORE "
1660 R=15
1670 GOSUB 1600
1680 J$="PLEASE TYPE IN YOUR NAM
E "
1690 R=20
1700 GOSUB 1600
1710 PRINT "BEST SCORE SO FAR ="
;HS::
1720 INPUT N$
1730 IF N$<>" " THEN 1780
1740 IF Z=1 THEN 1820
1750 PRINT "I SAID TYPE IN YOUR
NAME " ::
1760 Z=Z+1
1770 GOTO 1720
1780 PRINT :: "PRESS ANY KEY TO P
LAY" ::
1790 CALL KEY(O,K,S)
1800 IF S=0 THEN 1790
1810 GOTO 840
1820 PRINT :: "DON'T BOTHER THEN"

```

HOME COMPUTING WEEKLY 22 January 1985 Page 29



Grand prix



The race is on! Overtake all the competitors — but watch out for oil slicks — in Colin Ashworth's game for the TI-99/4A with Extended BASIC

Take your place behind the wheel for this fast-moving racing game for the TI-99/4A with Extended BASIC.

After the introduction, you will be asked to choose your skill level, which is from one to three. One is the easiest and three the hardest.

Now the screen clears and the track is set up. Your car appears at the bottom of the screen and you must overtake all the other cars — but watch out for oil slicks!

You must race as far as you can without crashing. If you find the game is too fast, alter the speeds in lines 740, 750 and 760.

If you crash, the screen clears and prints your score as well as the highest score to date.

How it works

100-170 define characters
190-210 define colours
220-230 set up screen
240-260 set up oil slicks
270-310 set up cars
320-440 move car
450-620 introduction
630-670 instructions
680-760 select skill level
770-830 crash routine, into cars or oil
840-890 crash into barrier routine
900 start of game
910-970 plays national anthem

Variables

I,B,C,D,E,F,G random position of cars
#1 to #4 racing cars
#5 your racing car
#6 #8 oil slicks

```
100 REM DEFINE CHARACTERS
110 CALL CHAR(33,"1A6F55CEBD5A6F
1A")
120 CALL CHAR(40,"FFFFFFFFFFFFFF
FF")
130 CALL CHAR(128,"1F101F01C2FCF
9C808E8F8F8EF011F1FF808F880433F9
F9390171F17F780F8F8")
140 CALL CHAR(132,"1F101F01C2FCF
BCA0BEAFBF8EF011F1FF808F880433FD
F53D057DF17F780F8F8")
150 CALL CHAR(136,"1F101F01C2FCF
9C909E8F8F8EF011F1FF808F880433FD
F53D0575F17F780F8F8")
160 CALL CHAR(140,"3C7EFFFFFFFFF7
E3C")
170 CALL CHAR(124,"34599D45B5AC7
CBF7EFF7EFFCA425149")
```



```

180 GOSUB 460
190 CALL COLOR(1,13,1)
200 CALL COLOR(2,16,1)
210 CALL CLEAR :: CALL SCREEN(4)
:: CALL MAGNIFY(3):: SCORE=0
220 REM SET UP SCREEN
230 CALL VCHAR(1,7,33,24):: CALL
VCHAR(1,26,33,24):: FOR I=8 TO
25 :: CALL VCHAR(1,I,40,24):: NE
XT I
240 REM SET UP OIL SLICKS
250 CALL SPRITE(#5,128,2,155,117
,0,0):: FOR I=1 TO 100 :: NEXT I
260 RANDOMIZE :: I=INT(RND*SPEED
4)+SPEED1 :: B=INT(RND*SPEED4)+S
PEED1 :: C=INT(RND*SPEED4)+SPEED
1
270 CALL SPRITE(#6,140,2,1,80,I,
0,#7,140,2,1,120,B,0,#8,140,2,1,
155,C,0)
280 REM SET UP CARS
290 RANDOMIZE :: D=INT(RND*SPEED
4)+SPEED1 :: E=INT(RND*SPEED4)+S
PEED1 :: F=INT(RND*SPEED4)+SPEED
1 :: G=INT(RND*SPEED4)+SPEED1
300 CALL SPRITE(#1,132,11,1,60,D
,0):: CALL SPRITE(#2,136,5,1,100
,E,0)
310 CALL SPRITE(#3,132,13,1,130,
F,0):: CALL SPRITE(#4,136,7,1,17
5,6,0)
320 REM MOVE CAR
330 CALL KEY(0,K,S):: CALL COINC
(ALL,R):: IF R=-1 THEN 770
340 CALL POSITION(#5,E,J):: IF J
<45 THEN 840 ELSE SCORE=SCORE+1
350 CALL POSITION(#5,E,J):: IF J
>190 THEN 840 ELSE SCORE=SCORE+1
360 IF SCORE>HSCORE THEN 370 ELSE
380
370 HSCORE=SCORE
380 IF S=-1 THEN 330
390 CALL COINC(ALL,C):: IF C=-1
THEN 770
400 IF S=0 THEN CALL MOTION(#5,0
,0):: GOTO 330
410 IF K<44 OR K>46 THEN 330
420 CALL MOTION(#5,0,(K-45)*SPEE
D5):: CALL COINC(ALL,C):: IF C=-
1 THEN 770 ELSE 330
430 CALL MOTION(#5,0,(K-45)*40):
: CALL COINC(ALL,C):: IF C=-1 TH
EN 770 ELSE 330
440 GOTO 330
450 REM INTRODUCTION
460 CALL CLEAR :: CALL SCREEN(16
)

```

```

470 DISPLAY AT(12,5):"G R A N D
P R I X" :: DISPLAY AT(13,5):"
*****"
480 DISPLAY AT(14,14):"B Y" :: D
ISPLAY AT(15,14):"***"
490 DISPLAY AT(16,8):"COLIN ASHW
ORTH" :: DISPLAY AT(17,8):"*****
*****"
500 FOR N=1 TO 4
510 CALL SOUND(60,220,8,-5,0)::
CALL SOUND(60,220,8,-5,5)
520 NEXT N
530 FOR F=1000 TO 2000 STEP 15
540 CALL SOUND(-99,111,30,111,30
,F,30,-8,0)
550 NEXT F
560 FOR F=1 TO 30
570 CALL SOUND(-99,111,30,111,30
,4000,30,-8,0)
580 NEXT F
590 CALL CLEAR :: FOR I=1 TO 400
:: NEXT I
600 DISPLAY AT(12,1):"DO YOU WAN
T INSTRUCTIONS?"
610 CALL KEY(0,K,S):: IF S=0 THE
N 600 :: IF K=89 THEN 640 ELSE 8
90
620 FOR I=1 TO 400 :: NEXT I
630 REM INSTRUCTIONS
640 CALL CLEAR :: DISPLAY AT(12,
1):"YOU HAVE TO TRAVEL AS FAR AS
YOU
CAN BEFORE YOU CRASH,"
650 DISPLAY AT(16,1):"AVOIDING O
THER CARS,OIL
SLICKS AND BARRIER
S."
660 DISPLAY AT(20,1):"USE THE '<
' AND '>' KEYS TO
MOVE."
670 DISPLAY AT(24,1):"PRESS ANY
KEY TO CONTINUE..." :: CALL KEY(
0,K,S):: IF S=0 THEN 670 :: IF K
<>1 THEN 890 ELSE 670
680 REM SKILL LEVEL
690 CALL CLEAR :: FOR I=1 TO 400
:: NEXT I :: DISPLAY AT(12,5):"
SKILL LEVEL [1 TO 3]"
700 CALL KEY(0,K,S):: IF S=0 THE
N 700
710 IF K<>49 THEN 720 ELSE 740
720 IF K<>50 THEN 730 ELSE 750
730 IF K<>51 THEN 700 ELSE 760
740 SPEED1=5 :: SPEED2=15 :: SPE
ED3=25 :: SPEED4=35 :: SPEED5=20
:: GOTO 190
750 SPEED1=10 :: SPEED2=20 :: SP

```



```

EED3=30 :: SPEED4=40 :: SPEED5=2
7 :: GOTO 190
760 SPEED1=15 :: SPEED2=25 :: SP
EED3=35 :: SPEED4=45 :: SPEED5=3
5 :: GOTO 190
770 CALL SOUND(1000,-5,0):: FOR
I=1 TO 9 :: CALL MOTION(#I,0,0):
: NEXT I :: CALL PATTERN(#5,124)
:: FOR A=1 TO 500 :: NEXT A
780 CALL CLEAR :: CALL SCREEN(16
):: CALL DELSPRITE(ALL)
790 REM CRASH
800 FOR I=1 TO 400 :: NEXT I ::
DISPLAY AT(12,4):"YOU CRASHED YO
UR CAR"
810 DISPLAY AT(16,5):"YOUR SCORE
IS:";SCORE
820 DISPLAY AT(20,8):"HIGH SCORE
:";HSCORE
830 DISPLAY AT(24,1):"PRESS SPAC
E BAR TO CONTINUE" :: CALL KEY(0
,K,S):: IF S=0 THEN 830 :: IF K=
32 THEN 690 ELSE 830
840 CALL SOUND(1000,-5,0):: CALL
CLEAR :: CALL DELSPRITE(ALL)::
CALL SCREEN(16):: FOR I=1 TO 400

```

```

:: NEXT I
850 REM CRASH
860 DISPLAY AT(12,1):"YOU RAN IN
TO THE HEDGES" :: DISPLAY AT(16,
5):"YOUR SCORE IS:";SCORE
870 DISPLAY AT(20,8):"HIGH SCORE
:";HSCORE
880 DISPLAY AT(24,1):"PRESS SPAC
E BAR TO CONTINUE" :: CALL KEY(0
,K,S):: IF S=0 THEN 880 :: IF K=
32 THEN 690 ELSE 880
890 CALL CLEAR :: FOR I=1 TO 400
:: NEXT I
900 DISPLAY AT(12,1):"THE RACE I
S ABOUT TO START"
910 REM NATIONAL ANTHEM
920 RESTORE 940
930 READ B,A :: IF A=0 THEN 970
:: CALL SOUND(A,B,0):: GOTO 930
940 DATA 392,300,392,300,440,300
,370,450,392,150,440,300
950 DATA 494,300,494,300,523,300
,494,450,440,150,392,300
960 DATA 440,300,392,300,370,300
,392,900,0,0
970 GOTO 690

```

UNLEASH THE POWER WITHIN YOUR '64

EXTENDED BASIC

With Built-in M/C
ASSEMBLER



£19.50

£21.50

51 EXTRA COMMANDS
10 EXTRA CTRL CODES
60 page REF. MANUAL
2 or 3 Pass ASSEMBLER

RENUMBER (inc. GOTOs & GOSUBs)
PROCEDURE & other structure commands
PERIPHERAL, MUSIC, GRAPHIC, SPRITE, HI-RES
DRAWING, 'F' KEY DEFINITION, and many more.
AND YET THIS 16K M/C PROG. IS 'HIDDEN', USING
JUST 2.5K OF BASIC MEMORY — LEAVING YOU:-

36.5K of USEABLE BASIC MEM.

ADVANCED CENTRONICS INTERFACE



**S/W ONLY
£9.50 + £2**



**S/W + CABLE
£19.50 + £2**

PROGRAM LISTINGS SHOW:-
ALL CBM 64 CONTROL CODES.
THE CBM 64 GRAPHICS SET.
USER DEFINED CHARACTERS.
SPACES (UPPER or LOWER).
COMPATIBILITY

Prog. will Auto Re-LOCATE at top of memory, or can be
LOADED into a MEMORY LOCATION as SPECIFIED BY
USER! VERY 'USER FRIENDLY'. YOU CAN 'MIX' HI &
LOW RES SCREEN DUMPS, EVEN DIRECT OFF DISC?

HYPERDRIVE THE ULTIMATE DISK UTILITY



By Dave
Partington

£9.50

FEATURES A TURBOCHARGER
(Make your disk-based S/W load 4 times faster!)
+ 28 EXTRA FACILITIES...

MAKE ERRORS 20-21-22-23-29 : CHECK ERRORS :
MAKE ALPHA NUMERIC DIRECTORY : RENAME DISK
— (inc. both IDs) : MAKE DIRECTORY NON LIST/NON
LOAD : UN-SCRATCH : REVEAL SCRATCH — (NAMES
EVERY FILE THAT HAS EVER BEEN SCRATCHED) 1
BYTE READ/WRITE : MAKE NON-REVERSABLE WRITE
PROTPLUS MANY MORE USEFUL FEATURES.

This Month's
PRINTER OFFER...

EPSON RX 80F/T

(R.R.P. £327 inc VAT)

YOU PAY THE Ex. VAT RRP £285
& WE'LL PAY THE VAT (£42.75)
DELIVERY (£10), CABLE & S/W,
(£19.50), & a free copy of
SUPER SCRAMBLE.

* Review in last weeks HCW.

WANTED!
C.16 SOFTWARE
PHONE US NOW!

PROGRAMMERS

We need ALL kinds of S/W for most micros.
S/W must be HIGH QUALITY, useful, user-
friendly and, preferably, innovative. If you
fit the bill — please contact us.

SUPER SCRAMBLE

One of our 'CLASSIC'
Arcade Games



£6.50

HERE'S WHAT THE REVIEWERS HAVE TO SAY SO FAR:-
Simply, it is even better than the original arcade game... it
is an example of state of the arts game.

YOUR COMMODORE

Currently my favourite variant of the game.

COMMODORE

The design is excellent with smooth scrolling and nicely
animated radar scanners... If you need a Scramble to
complete your collection, this one is worth a look.

HOME COMPUTER WEEKLY

ALL PRICES INCLUDE VAT,
POST & PACKING.
OVERSEAS PLEASE ADD £1:
TRADE ENQS WELCOME

MUSHROOM
01-670 3533



SOFTWARE
193 ROMMANY ROAD
LONDON SE27 9PR

It's you against your Texas in this Reversi-type game. Who's the smarter, you or it?
By C Tubb

This game is an old favourite, now on your TI. It's a version of Reversi, in which you capture the other player's counters, as you would in draughts.

However, instead of removing the counters from the board, they stay on, but change colour and become yours.

You take the blue counters and play against the computer. But don't try cheating — the computer is on the ball and will soon put a stop to your foul play!

How it works

120-140 zero's flags
 150-210 scans keyboard, goes to instructions if required
 230-250 resets flags to zero
 260 GOSUB to define colours, characters and draw board
 270 GOSUB to create moves table into BDS
 280 goes to players move entry routine
 290 clears prompt line
 300-320 checks if selected square is occupied
 340 if move successful then GOTO remove from moves table routine
 350-880 move pieces routine — reads moves from moves table and checks each one to find a legal move
 490-520 clears old pointer arrows, prints new ones at each location as computer

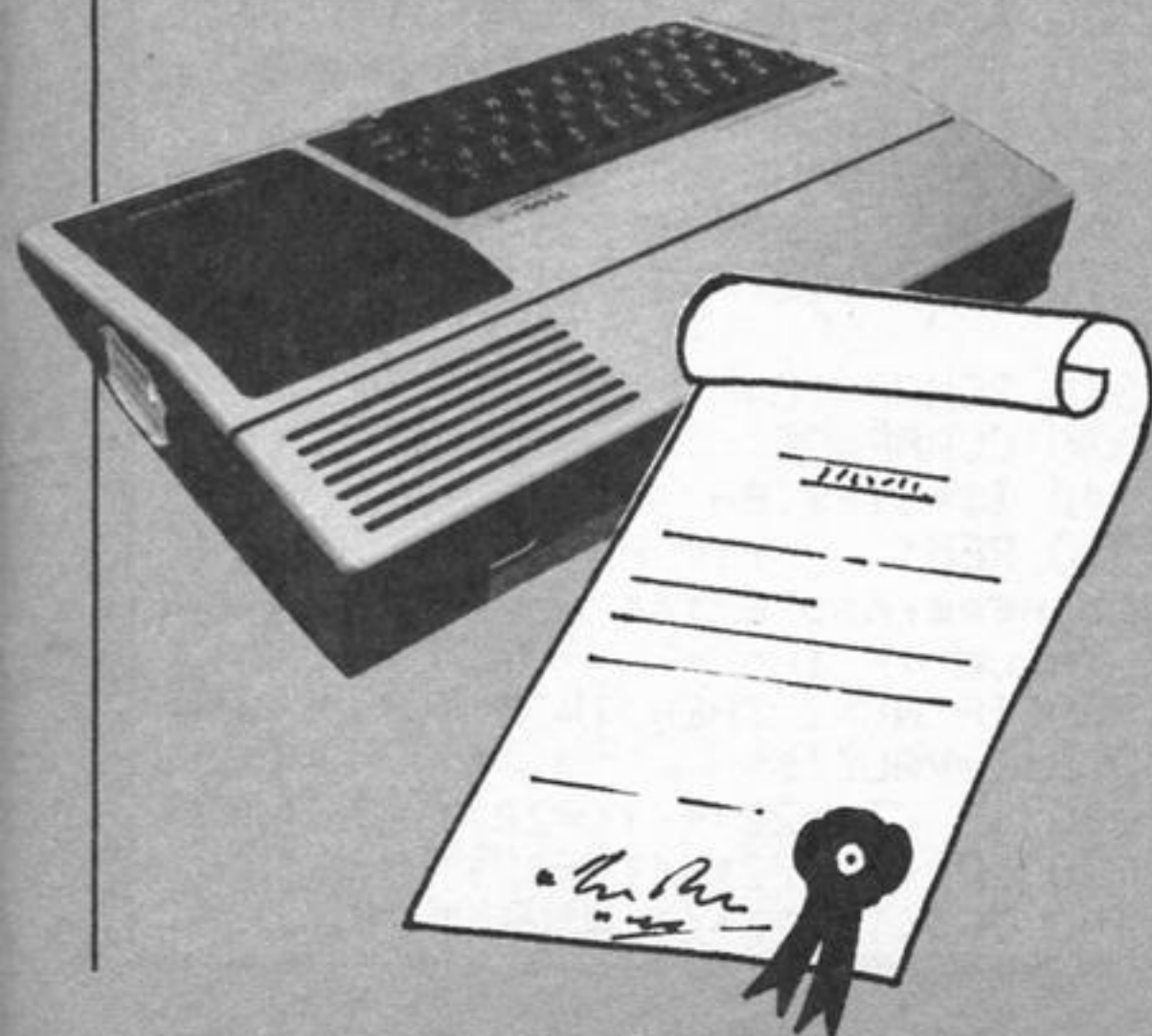
Pit your wits against your micro

620 finds what is in selected square
 700-740 prints counter in selected square, changes colour of taken counters
 750 sound of counters moving
 760 clears prompt line
 770-880 remainder of moves routine
 890 I can't move routine
 910 checks if moves table is empty. If so GOTO I can't move routine
 920-930 makes Oh! Oh! sound
 1000 clears prompt line
 1010-1020 clears pointer arrows
 1070 checks if square selected. If not GOTO your move routine
 1080 if move OK GOTO delete move from moves table
 1090 players move entry routine
 1190 when computer can't move, enters here for further move by player
 1250-1260 converts key entry
 1270 error check
 1280 clears prompt line
 1300 prints x arrow
 1320-1330 assigns counters into variables
 1360 if flag ND is 0 then check if new data is to be added to moves table
 1370-1400 checks where moves have progressed to
 1420 checks if end of game has been reached
 checks them
 570-580 scans adjoining squares for possible moves

1450-1470 creates moves table into BDS
 1480-1490 zero's flags
 1510-1520 zero's score registers
 1530 end of game routine
 1990-2000 delay
 2010 GOTO start of new game
 2020 defines characters and colours
 2230 draws board

2420-2570 prints two blue/two yellow counters
 2590-2680 deletes move from table
 2690-2750 illegal move sound, clears pointer arrows for further try by player
 2770 adds extra moves into moves table (BDS)
 2870 instructions

```
100 REM  *NO CALLS ARE MADE TO
ANY OF THE REM LINES AND THESE C
AN BE OMITTED IF REQUIRED.*
110 REM  *
120 RD=0
130 ND=0
140 IM=0
150 CALL CLEAR
160 PRINT "*****"
*****"::"
      "::::*****"
*****"::::::::::
170 PRINT "WOULD YOU LIKE INSTRU
CTIONS?"::"      (Y/N)"::"(
ENSURE ALPHA LOCK IS DOWN)"
180 CALL KEY(0,KEY,S)
190 IF S=0 THEN 180
200 REM  *CALL TO INSTRUCTIONS*
210 IF (KEY=89)+(KEY=121) THEN 28
60
220 CALL CLEAR
230 RD=0
240 ND=0
250 IM=0
260 GOSUB 2030
270 GOSUB 1450
280 GOTO 1090
290 CALL HCHAR(22,1,32,32)
300 CALL GCHAR(A,B,X)
310 REM  *IS IT A LEGAL MOVE?*
320 IF (X<>96)*(X<>104) THEN 2820
330 REM  IF IT IS-REMOVE FROM MO
VES TABLE*
340 GOSUB 2590
350 M$="  L(MY MOVE)L  "
360 FOR P=1 TO LEN(M$)
370 CALL HCHAR(22,P+10,ASC(SEG$(
```




```

M$,P,1)),1)
380 NEXT P
390 BCHAR=96
400 CCHAR=104
410 H=0
420 REM *MOVE MEN AND FIND HOW
MANY CAN BE TAKEN.*
430 FOR K=1 TO LEN(BD$)STEP 3
440 AA=VAL(SEG$(BD$,K,1))
450 BB=VAL(SEG$(BD$, (K+1), 1))
460 I$=STR$(AA)&STR$(BB)&". "
470 A=((AA+3)*2)-2
480 B=((BB+5)*2)-2
490 CALL VCHAR(5,9,32,16)
500 CALL HCHAR(4,10,32,16)
510 CALL HCHAR(A,9,37,1)
520 CALL HCHAR(4,B,36,1)
530 CALL GCHAR(A,B,Z)
540 IF Z<>112 THEN 880
550 CALL GCHAR(A,B,X)
560 IF (X<>112) THEN 2840
570 FOR C=-2 TO 2 STEP 2
580 FOR D=-2 TO 2 STEP 2
590 E=0
600 F=A
610 G=B
620 CALL GCHAR(F+C,G+D,Z)
630 IF Z<>BCHAR THEN 680
640 E=1
650 F=F+C
660 G=G+D
670 GOTO 620
680 CALL GCHAR(F+C,G+D,Z)
690 IF (Z<>CCHAR)+(E=0) THEN 820
700 CALL HCHAR(F,G,CCHAR,1)
710 CALL HCHAR(F-1,G,CCHAR+3,1)
720 IM=1
730 CALL HCHAR(F,G+1,CCHAR+1,1)
740 CALL HCHAR(F-1,G+1,CCHAR+2,1)
)
750 CALL SOUND(1,440,7)
760 CALL HCHAR(22,1,32,28)
770 IF (A=F)*(B=G) THEN 820
780 F=F-C
790 G=G-D
800 H=1
810 GOTO 700
820 NEXT D
830 NEXT C
840 IF (BCHAR=104)+(H=1) THEN 101
0
850 CALL VCHAR(5,9,32,16)
860 CALL HCHAR(4,10,32,16)
870 CALL GCHAR(A,B,Z)
880 NEXT K
890 REM *CANNOT MOVE*
900 RD=0

```

```

910 IF LEN(BD$)=3 THEN 1480
920 CALL SOUND(400,150,8)
930 CALL SOUND(400,110,8)
940 M$="I CAN'T MOVE.."
950 FOR P=1 TO LEN(M$)
960 CALL HCHAR(22,P+11,ASC(SEG$(
M$,P,1)),1)
970 NEXT P
980 FOR R=1 TO 500
990 NEXT R
1000 CALL HCHAR(22,4,32,28)
1010 CALL VCHAR(5,9,32,16)
1020 CALL HCHAR(4,10,32,16)
1030 IF BCHAR=104 THEN 290
1040 IF RD=0 THEN 1090
1050 CALL GCHAR(A,B,X)
1060 REM *IS SQUARE CLEAR?*
1070 IF (X<>96)*(X<>104) THEN 109
0
1080 GOSUB 2590
1090 M$="D(YOUR MOVE)D"
1100 RD=1
1110 FOR P=1 TO LEN(M$)
1120 CALL HCHAR(22,P+10,ASC(SEG$(
M$,P,1)),1)
1130 NEXT P
1140 CALL KEY(0,KEY,S)
1150 IF S=0 THEN 1140
1160 AA=KEY-64
1170 A=((KEY-61)*2)-2
1180 IF (I$="44.")*(KEY=48) THEN
1510
1190 REM *I CANNOT MOVE*
1200 IF KEY=48 THEN 350
1210 IF (A<6)+(A>20) THEN 2690
1220 CALL HCHAR(A,9,37,1)
1230 CALL KEY(0,KEY,S)
1240 IF S=0 THEN 1230
1250 B=((KEY-43)*2)-2
1260 BB=KEY-48
1270 IF (B<10)+(B>24) THEN 2690
1280 CALL HCHAR(22,4,32,28)
1290 CALL GCHAR(A,B,X)
1300 CALL HCHAR(4,B,36,1)
1310 IF X=112 THEN 1320 ELSE 269
0
1320 BCHAR=104
1330 CCHAR=96
1340 I$=STR$(AA)&STR$(BB)&". "
1350 REM *IF MOVES HAVE REAC
HED HERE,ADD EXTRA MOVES TO MOVE
S TABLE IF IM=0*
1360 IF ND=1 THEN 1420
1370 T=VAL(I$)
1380 IF (T>=22)*(T<=27) THEN 2770
1390 IF (T>=72)*(T<=77) THEN 2770
1400 IF (T=32)+(T=42)+(T=52)+(T=

```



```

62)+(T=37)+(T=47)+(T=57)+(T=67)T
HEN 2770
1420 IF (KEY=48)*(K)=LEN(BD$))TH
EN 1510
1430 GOTO 550
1440 REM MOVES DATA **DO NOT AT
TEMPT TO PUT ALL DATA INTO ONE S
TRING**
1450 BD1$="63.66.33.36.64.65.35.
34.43.56.46.53.14.15.84.85.41.48
.51.58.23.26.73.76."
1460 BD2$="32.37.62.67.24.25.74.
75.42.47.52.57.12.17.82.87.21.28
.71.78.22.27.72.77.44.44."
1470 BD$=BD1$&BD2$
1480 ND=0
1490 RD=0
1500 RETURN
1510 YOU=0
1520 ME=0
1530 CALL HCHAR(22,1,32,32)
1540 M$="END OF THE GAME.."
1550 FOR P=1 TO 17
1560 CALL HCHAR(22,P+7,ASC(SEG$(
M$,P,1)))
1570 NEXT P
1580 FOR X=6 TO 20 STEP 2
1590 FOR Y=10 TO 25 STEP 2
1600 CALL GCHAR(X,Y,S)
1610 IF S=104 THEN 1630
1620 IF S=96 THEN 1650
1630 ME=ME+1
1640 GOTO 1660
1650 YOU=YOU+1
1660 NEXT Y
1670 NEXT X
1680 CALL HCHAR(21,1,32,32)
1690 CALL HCHAR(23,1,32,32)
1700 CALL HCHAR(24,1,32,32)
1710 C$=STR$(ME)
1720 Y$=STR$(YOU)
1730 CALL HCHAR(22,1,32,32)
1740 IF YOU>ME THEN 1870
1750 IF YOU<ME THEN 1820
1760 CALL HCHAR(22,1,32,32)
1770 M$="IT'S A DRAW!!"
1780 FOR P=1 TO 13
1790 CALL HCHAR(22,P+11,ASC(SEG$(
M$,P,1)))
1800 NEXT P
1810 GOTO 1910
1820 M$="I WIN!!"
1830 FOR P=1 TO 7
1840 CALL HCHAR(22,P+13,ASC(SEG$(
M$,P,1)))
1850 NEXT P
1860 GOTO 1910

```

```

1870 M$="YOU WIN!"
1880 FOR P=1 TO 8
1890 CALL HCHAR(22,P+13,ASC(SEG$(
M$,P,1)))
1900 NEXT P
1910 M$="YOU SCORED:"&Y$
1920 FOR P=1 TO LEN(M$)
1930 CALL HCHAR(23,P+9,ASC(SEG$(
M$,P,1)))
1940 NEXT P
1950 M$="I SCORED:.."&C$
1960 FOR P=1 TO LEN(M$)
1970 CALL HCHAR(24,P+9,ASC(SEG$(
M$,P,1)))
1980 NEXT P
1990 FOR I=1 TO 2000
2000 NEXT I
2010 GOTO 220
2020 REM *DEFINE CHARACTERS
AND COLORS*
2030 CALL CLEAR
2040 CALL CHAR(96,"7F7F7F7F7F7F7
F00")
2050 CALL CHAR(97,"FEFEFEFEFEFEFE
E00")
2060 CALL CHAR(98,"00FEFEFEFEFEFE
EFE")
2070 CALL CHAR(99,"007F7F7F7F7F7
F7F")
2080 CALL CHAR(100,"007E7E7E7E7E
7E00")
2090 CALL CHAR(104,"7F7F7F7F7F7F
7F00")
2100 CALL CHAR(105,"FEFEFEFEFEFE
FE00")
2110 CALL CHAR(106,"00FEFEFEFEFE
FEFE")
2120 CALL CHAR(107,"007F7F7F7F7F
7F7F")
2130 CALL CHAR(108,"007E7E7E7E7E
7E00")
2140 CALL CHAR(36,"004163777F3E1
C08")
2150 CALL CHAR(37,"00783C1E0F1E3
C78")
2160 CALL CHAR(112,"808080808080
80FF")
2170 CALL CHAR(113,"010101010101
01FF")
2180 CALL CHAR(114,"FF0101010101
0101")
2190 CALL CHAR(115,"FF8080808080
8080")
2200 CALL COLOR(9,5,2)
2210 CALL COLOR(10,11,2)
2220 CALL COLOR(11,2,8)
2230 REM SET UP BOARD

```


TI-99/4A PROGRAM

```

2240 FOR I=6 TO 20 STEP 2
2250 FOR J=10 TO 24 STEP 2
2260 CALL HCHAR(I-1,J,115,1)
2270 CALL HCHAR(I,J,112,1)
2280 CALL HCHAR(I-1,J+1,114,1)
2290 CALL HCHAR(I,J+1,113,1)
2300 NEXT J
2310 NEXT I
2320 X=0
2330 FOR J=6 TO 20 STEP 2
2340 X=X+1
2350 CALL HCHAR(J,8,X+64,1)
2360 NEXT J
2370 X=0
2380 FOR J=10 TO 24 STEP 2
2390 X=X+1
2400 CALL HCHAR(3,J,X+48,1)
2410 NEXT J
2420 CALL HCHAR(12,18,96,1)
2430 CALL HCHAR(11,18,99,1)
2440 CALL HCHAR(12,19,97,1)
2450 CALL HCHAR(11,19,98,1)
2460 CALL HCHAR(14,16,96,1)
2470 CALL HCHAR(13,16,99,1)
2480 CALL HCHAR(14,17,97,1)
2490 CALL HCHAR(13,17,98,1)
2500 CALL HCHAR(12,16,104,1)
2510 CALL HCHAR(11,16,107,1)
2520 CALL HCHAR(12,17,105,1)
2530 CALL HCHAR(11,17,106,1)
2540 CALL HCHAR(14,18,104,1)
2550 CALL HCHAR(13,18,107,1)
2560 CALL HCHAR(14,19,105,1)
2570 CALL HCHAR(13,19,106,1)
2580 RETURN
2590 REM *REMOVE OCCUPIED SQUARE LOCATIONS FROM MOVES TABLE*
2600 RD=1
2610 CALL SOUND(10,1000,5)
2620 LS=LEN(BD$)
2630 XX=POS(BD$,I$,1)
2640 IF XX=0 THEN 2680
2650 EX1$=SEG$(BD$,1,XX-1)
2660 EX2$=SEG$(BD$,XX+3,LS-3)
2670 BD$=EX1$&EX2$
2680 RETURN
2690 CALL SOUND(200,550,1,440,1,330,1)
2700 CALL HCHAR(4,10,32,16)
2710 CALL VCHAR(5,9,32,16)
2720 CALL HCHAR(22,1,32,32)
2730 FOR I=1 TO 300
2740 NEXT I
2750 GOTO 1090
2760 IF (RD=0)+(IM=0) THEN 1090
2770 REM **EXTRA MOVES DATA.DO

```

```

NOT ATTEMPT TO INCLUDE THIS WITH
THE OTHER PART OF BD$**
2780 IF ND=1 THEN 1420
2790 LET BD$="11.18.81.88.83.86.
61.68.31.38.16.13."&BD$
2800 ND=1
2810 GOTO 1420
2820 CALL SOUND(200,550,1,440,1,330,1)
2830 GOTO 1090
2840 CALL SOUND(200,550,1,440,1,330,1)
2850 GOTO 1090
2860 CALL CLEAR
2870 PRINT "
THE OBJECT
T IS TO CAPTURE THE COMPUTERS ME
N ASIN DRAUGHTS. IN THIS GAME HOW
"
2880 PRINT "-EVER, THE MEN REMAIN
ON THE BOARD BUT CHANGE COLOUR
AND BECOME YOURS."
2890 PRINT " ":"THE WINNER IS TH
E ONE WITH MOST MEN WHEN THE GA
ME ENDS."
2900 PRINT " ":"YOU PLAY FIRST A
ND ARE BLUE. THE COMPUTER WILL NO
T ALLOW YOU TO MAKE A WRONG MOVE
."
2910 PRINT "IF YOU CANNOT MOVE E
NTER 0 AND THE COMPUTER WILL PL
AY AGAIN."
2920 PRINT "ENTER MOVES AS:LETTE
R=NUMBER"
2930 PRINT " ":"PRESS ANY KEY T
O START>"
2940 CALL KEY(0,KEY,S)
2950 IF S=0 THEN 2940
2960 GOTO 220

```



This singalong program for the TI-99/4A will play Happy Birthday whilst showing the cake and candles on screen.

You might choose to use it as a birthday surprise and ask those celebrating to blow out the electronic candles on screen.

After a few blows and when everyone is going blue in the face, a press of any key will do the job for you.

TI-99/4A PROGRAMS

Blow away

**Celebrate our birthday with
this sound and graphic
program from 15 year old
James Wright**

```

6 REM THIS PROGRAM IS FOR THE TI99/4A
7 REM BY JAMES, D, WRIGHT
8 REM "HAPPY BIRTHDAY"
10 CALL CLEAR
20 CALL SCREEN(2)
30 CALL CHAR(64, "FFFFFFFFFFFFFFFFFFFF")
40 CALL COLOR(5, 5, 2)
50 CALL VCHAR(5, 16, 64, 5)
60 CALL VCHAR(5, 13, 64, 5)
70 CALL VCHAR(5, 19, 64, 5)
80 CALL CHAR(112, "18183C3C3C3C7E7E7E7E")
90 CALL COLOR(11, 11, 2)
100 CALL HCHAR(4, 16, 112)
110 CALL HCHAR(5, 16, 113)
120 CALL CHAR(113, "FFFFFF7E7E3C3C1818")
130 CALL HCHAR(4, 13, 112)
140 CALL HCHAR(5, 13, 113)
150 CALL HCHAR(4, 19, 112)
160 CALL HCHAR(5, 19, 113)
170 CALL CHAR(152, "FFFFFFFFFFFFFFFFFFFF")
180 CALL COLOR(16, 16, 2)
190 CALL HCHAR(10, 11, 152, 11)
200 FOR I=1 TO 8
210 CALL COLOR(I, 9, 1)
220 NEXT I
230 CALL HCHAR(18, 4, 104)
240 CALL HCHAR(11, 10, 152, 13)
250 CALL HCHAR(12, 9, 152, 15)
260 CALL HCHAR(13, 9, 152, 15)
270 CALL HCHAR(14, 9, 152, 15)
280 CALL HCHAR(15, 9, 152, 15)
290 CALL CHAR(40, "FFFFFFFFFFFFFFFFFFFF")
300 CALL HCHAR(16, 8, 40, 17)
310 CALL HCHAR(17, 9, 40, 15)
320 CALL COLOR(2, 5, 2)
330 CALL CHAR(41, "FF7F3F1F0F070301")
340 CALL HCHAR(17, 8, 41)
350 CALL HCHAR(16, 7, 41)
360 CALL CHAR(42, "FFFEFCF8F0E0C080F")
370 CALL HCHAR(17, 24, 42)
380 CALL HCHAR(16, 25, 42)
390 CALL CHAR(153, "0103070F1F3F7FFF")
400 CALL HCHAR(10, 10, 153)
410 CALL HCHAR(11, 9, 153)
420 CALL CHAR(154, "80C0E0F0F8FCFEFFF")
430 CALL HCHAR(10, 22, 154)
440 CALL HCHAR(11, 23, 154)
441 CALL CHAR(56, "8142241818244281")
442 CALL COLOR(4, 5, 16)
443 CALL HCHAR(18, 1, 56, 224)
450 CALL SOUND(300, 262, 0)
460 CALL SOUND(300, 262, 0)
470 CALL SOUND(600, 294, 0)
480 CALL SOUND(600, 262, 0)
490 CALL SOUND(600, 349, 0)

```

```

500 CALL SOUND(1200, 330, 0)
520 CALL SOUND(300, 262, 0)
530 CALL SOUND(300, 262, 0)
540 CALL SOUND(600, 294, 0)
550 CALL SOUND(600, 262, 0)
560 CALL SOUND(600, 392, 0)
570 CALL SOUND(1200, 349, 0)
580 CALL SOUND(300, 262, 0)
590 CALL SOUND(300, 262, 0)
600 CALL SOUND(600, 262, 0)
610 CALL SOUND(600, 440, 0)
620 CALL SOUND(600, 349, 0)
630 CALL SOUND(600, 330, 0)
640 CALL SOUND(600, 294, 0)
650 CALL SOUND(300, 494, 0)
660 CALL SOUND(300, 494, 0)
670 CALL SOUND(600, 440, 0)
680 CALL SOUND(600, 349, 0)
690 CALL SOUND(600, 392, 0)
700 CALL SOUND(1200, 349, 0)
720 CALL SOUND(300, 262, 0)
730 CALL SOUND(300, 262, 0)
740 CALL SOUND(600, 294, 0)
750 CALL SOUND(600, 262, 0)
760 CALL SOUND(600, 349, 0)
770 CALL SOUND(1200, 330, 0)
790 CALL SOUND(300, 262, 0)
800 CALL SOUND(300, 262, 0)
810 CALL SOUND(600, 294, 0)
820 CALL SOUND(600, 262, 0)
830 CALL SOUND(600, 392, 0)
840 CALL SOUND(1200, 349, 0)
860 CALL SOUND(300, 262, 0)
870 CALL SOUND(300, 262, 0)
880 CALL SOUND(600, 262, 0)
890 CALL SOUND(600, 440, 0)
900 CALL SOUND(600, 349, 0)
910 CALL SOUND(600, 330, 0)
920 CALL SOUND(600, 294, 0)
930 CALL SOUND(300, 494, 0)
940 CALL SOUND(300, 494, 0)
950 CALL SOUND(600, 440, 0)
960 CALL SOUND(600, 349, 0)
970 CALL SOUND(600, 330, 0)
980 CALL SOUND(1200, 349, 0)
981 CALL KEY(0, K, S)
982 IF S=0 THEN 981
983 IF K=30 THEN 984
984 CALL COLOR(11, 2, 2)
985 CALL COLOR(11, 11, 2)
986 CALL COLOR(11, 2, 2)
987 CALL COLOR(11, 11, 2)
988 CALL COLOR(11, 2, 2)
990 FOR DELAY=1 TO 9000000000000000
1000 NEXT DELAY

```


C5 program

I thought that some of the readers of your excellent magazine would be interested in this program, which I have written for the C5 micro carputer. It innovates all the most useful items included in this excellent vehicle/state-board which Sir Clive has introduced to endanger the lives of all those unfortunate enough to have purchased one of these toys.

```
10 LET SPEED=0
20 LET WEIGHT=50
30 IF IKEYS = "KEY
  INSERTED" THEN
  GOTO 50
40 GOTO 30
50 IF INKEYS="ACCEL-
  ERATOR ON" THEN
  GOTO 70
60 GOTO 50
70 LET SPEED=SPEED+1
80 PRINT SPEED
90 LET DISTANCE = DIS-
  TANCE + SPEED
100 IF DISTANCES > 10
```

THEN PRINT "YOU
HAVE TRAVELLED
OVER HALF THE
DISTANCE"

```
110 IF DISTANCE = 20
  THEN GOTO 140
120 IF SPEED = 15 THEN
  GOTO 80
130 GOTO 50
140 LET SPEED = 0
150 PRINT "OUT OF POW-
  ER"
160 IF INKEYS = "BAT-
  TERY RECHARGED"
  THEN GOTO 10
170 PAUSE HALF AN
  HOUR
180 GOTO 160
```

On another tack, why must you publish so many TI-99/4A owners' letters? In issue 100 there were five TI letters out of a total of eight letters. Don't you think we other users get a bit bored of reading how "good" the TI is and how bad the other computers are?

M Graham, Congleton

Indiana moans

I am writing to express my views on US Gold's Indiana Jones in the Lost Kingdom. US Gold's software is usually first class, but this one is a long way from their normal standard.

The game offers six — don't swoon at this large amount — screens, all of which are poor and offer only limited graphics. Indy himself waddles like a duck and is extremely difficult to move.

I only hope that if people want to buy it it's because of the game and not the Indiana Jones' connection. Having said all this, the tune which is played is Indy's theme tune, and that at least is very nicely done.

T Simcox, Allenton

Problems, problems

Help! I have recently bought Daley Thompson's Decathlon. It's a very good game when it works.

I went down to my local stockist with my £7, all ready to buy Decathlon. In the shops were several copies, but would any of them work? No. Not one copy would load, or if the first side loaded the second side wouldn't. After attempting three or four copies I gave up and went to another store. Again I tried several copies but they wouldn't work either. I tried one last copy, and it worked, both sides.

I bought it straight away and rushed home to try it. It didn't work. I tried several times until it worked. Now every time I want to play it I have to load it several times before I can get it to work, and then sometimes side two doesn't work. My friends also have the same trouble.

Paul Chatwin, Broadstairs

Match Day winner...

Andrew Musgrove (HCW 100) wanted to know if anyone had thrashed the computer on Match Day. I can proudly boast that I have — with a 5-1 victory on the international level being my best result — but I've also scored eight goals on the amateur level.

I quite agree with Andrew's comments and any football fan should buy this program, which is the best sports simulation I've seen.

Paul Grant, Northampton

Sharpen up

Here are some corrections to Blow away, for the Texas, printed in HCW 100. These corrections mean that the music is no longer flat.

Line 600 should read: CALL SOUND (600,523,0)
Line 650: CALL SOUND (300,466,0)

Line 660: CALL SOUND (300,466,0)

Line 880: CALL SOUND (600,523,0)

Line 930: CALL SOUND (300,466,0)

Line 940: CALL SOUND (300,466,0)

Line 970: CALL SOUND (600,392,0)

I S Goldman, Wirral

...and another

I am writing with reference to Andrew Musgrove's letter about Match Day, by Ocean. I have managed to thrash the computer 9-0 at amateur level and 7-0 at both professional and international level, five minutes each half.

There are two bugs I've noticed. Firstly, when the players come on the pitch (close-up shot) the pitch is green. If you change the pitch colour and start up a new match the close-up shows the pitch still green.

Also, when the clock is just turning to 90, kick the ball off the field. If you do it just right you should get an extra 55 minutes.

Neil MacLennan, Canvey Island

THERE'S A GHOST IN MY HOUSE



Spooks and spectres are out to scare you to death in Daryl Coates's program. You're locked in with phantoms of the night

In this game you must get the key from the top right hand corner and open the green door in the bottom left corner. This means you can move on to the next screen.

But it's not as easy as it sounds — ghosts will try to scare you out of your wits, and if you're frightened three times you're a goner. On each screen — there are 15 in all — the ghosts move faster. And don't forget that they can move through walls, while you can't.

Move on to each floor level by standing underneath a rope and moving up or down. Use the arrow keys for movement.

The game includes an unusual feature, in line 410. It's the address of the GPL register or sprite information byte. If this value is ANDed with 32 equals 32, then there is a coincidence. This is much faster and more reliable than CALL COINC

(ALL,CH)::IF CH THEN 4400 — try it and see.

Variables

X1,Y1 position of sprite number 1
A general loops or variables
SKILL level
LIVES number of scares

How it works

20-110 define chars, call instructions, print scores, lives etc
120-220 set up screen
230-250 define ghosts, man
260-430 main game loop
440-490 next screen
5000-5500 move left
560-610 move right
620-650 move up
660-690 move down
700-750 loss of life, end of game
760-910 procedure for instructions: replace this with GOSUB or PROC

Note: Wherever the words HAUNTED HOUSE appear, type in, while holding the CTRL key, ABCDEFG and AHC-IF. All will be revealed when you run the program.

10 REM HAUNTED HOUSE

(c) D.M. COATES

23/5/84

20 CALL CLEAR :: CALL SCREEN(2):
: CALL MAGNIFY(3)

30 FOR A=1 TO 14 :: CALL COLOR(A
,16,1):: NEXT A

40 CALL CHAR(40,"007F7F7F00F7F7F
7"):: CALL COLOR(2,7,11)

50 CALL CHAR(59,"0305050B0817171
7171329010302020EC0A0A0D010E8E8E
8E8C89480C0404070")

60 CALL CHAR(129,"44447C447C4444
441028447C447C444444444444444C5464
3A4464544C64544C44")

70 CALL CHAR(133,"7C282828282828
287C50505C505C507C704854644C5468
70384C54644C546438")

80 CALL CHAR(137,"384C4838142464
38"):: CALL COLOR(13,16,7,14,16,
7):: CALL CHAR(58,"0000000000003
030")

90 CALL INSTRUCTIONS

100 CALL CHAR(34,"38444438101C10
1C"):: CALL CHAR(112,"FFFFFFFFF
FFFFFF"):: CALL COLOR(11,13,1)

110 DISPLAY AT(1,1):"YOUR SCORE~
0": "HIGH SCORE~ 0"

:: DISPLAY AT(1,21):"LEVEL~ 1" :
: DISPLAY AT(2,21):"SCARE~ 0"

120 CALL HCHAR(3,1,40,32):: CALL
VCHAR(1,1,40,24):: CALL VCHAR(1
,32,40,24):: CALL HCHAR(24,1,40,
32)

130 CALL HCHAR(23,1,40,32):: CAL
L VCHAR(21,1,112,2):: CALL HCHAR
(6,31,34)

140 DISPLAY AT(23,8)SIZE(13):"HA
UNTED(HOUSE"

150 CALL VCHAR(3,13,40,5):: CALL
VCHAR(3,14,40,5):: CALL VCHAR(7
,20,40,5):: CALL VCHAR(7,21,40,5


```

)
160 CALL VCHAR(15,15,40,8):: CAL
L VCHAR(15,16,40,8)
170 CALL VCHAR(19,22,40,4):: CAL
L VCHAR(19,23,40,4)
180 CALL CHAR(33,"18181818181818
18"):: CALL VCHAR(4,3,33,7):: CA
LL VCHAR(4,11,33,19):: CALL VCHA
R(4,16,33,7):: CALL VCHAR(4,25,3
3,7)
190 CALL VCHAR(8,28,33,7):: CALL
VCHAR(12,19,33,11):: CALL VCHAR
(16,3,33,7):: CALL VCHAR(16,27,3
3,7)
200 FOR A=1 TO 13 :: READ X,Y,Z
:: CALL HCHAR(X,Y,40,Z):: NEXT A
210 DATA 7,5,5,7,18,6,7,27,5,11,
2,8,11,13,14,11,30,2,15,2,8,15,1
3,5,15,21,11
220 DATA 19,5,5,19,13,5,19,21,5,
19,29,3
230 CALL CHAR(96,"000101037F3F63
030707030101000000F028ECFCF8FOC0
C0C0E0E0C0C0E07000")
240 CALL CHAR(100,"0F14373F1FOFO
3030307070303070E000008080C0FEFC
6C0E0E0C080800000000")
250 CALL CHAR(104,"0305050B08171
717171329010302020EC0A0A0D010E8E
8E8E8C89480C0404070")
260 REM *****THE GAME*****
270 SCORE=-100 :: LIVES=0 :: DIS
PLAY AT(2,27):USING "##":LIVES
280 SKILL=0 :: GOTO 440
290 CALL SPRITE(#2,100,16,28,9,0
,.6*(SKILL+2),#3,96,16,60,233,0,
-.7*(SKILL+2))
300 CALL SPRITE(#1,104,13,161,22
9)
310 CALL SPRITE(#4,100,16,92,9,0
,.5*(SKILL+2),#5,96,16,125,233,0
,-.6*(SKILL+2),#6,100,16,157,9,0
,.8*(SKILL+2))
320 HISCORE=MAX(SCORE,HISCORE)::
DISPLAY AT(1,12)SIZE(8):USING "
#####":SCORE :: DISPLAY AT(2,
12)SIZE(8):USING "#####":HISC
ORE
330 CALL PEEK(-31877,A)
340 CALL POSITION(#1,X,Y):: X1=(
X+15)/8 :: Y1=(Y+11)/8
350 CALL GCHAR(X1,Y1,CH)
360 CALL KEY(1,K,S):: IF S=0 THE
N 410
370 IF K=2 THEN 500
380 IF K=3 THEN 560

```

```

390 IF K=0 AND CH=33 THEN 660
400 IF K=5 AND CH=33 THEN 620
410 CALL PEEK(-31877,A):: IF (A
AND 32)=32 THEN 700
420 GOTO 340
430 REM SUBROUTINES FOR
MOVEMENT,LOSS OF
LIFE AND NEXT SCREEN
440 SCORE=SCORE+100 :: SKILL=SKI
LL+1 :: DISPLAY AT(1,27):USING "
##":SKILL :: IF SKILL=16 THEN 45
0 ELSE 290
450 FOR A=0 TO 30 STEP 2 :: CALL
SOUND(-99,262,A,330,A,392,A)::
CALL SOUND(-99,262,A,392,A,523,A
)
460 CALL SOUND(-99,523,A,659,A,7
84,A):: NEXT A
470 DISPLAY AT(23,1):" WELL DO
NE YOU HAVE JUST COMPLETED AL
L 15 SCREENS " :: FOR A=1 TO 800
:: NEXT A
480 DISPLAY AT(23,1):" BUT YOU H
AVE BEEN SENTENCED TO DO THEM
ALL AGAIN " :: FOR A=1 TO 80
0 :: NEXT A :: CALL HCHAR(23,1,4
0,64)
490 DISPLAY AT(23,8)SIZE(13):"HA
UNTED(HOUSE" :: GOTO 280
500 CALL GCHAR(X1,Y1-1,CH):: IF
CH=40 THEN 410
510 SCORE=SCORE+10
520 IF CH=112 AND FLAG THEN 530
ELSE IF CH=112 THEN 410 ELSE 540
530 FOR A=440 TO 880 STEP 10 ::
CALL SOUND(-99,A,0):: NEXT A ::
GOTO 440
540 CALL MOTION(#1,0,-50)
550 CALL MOTION(#1,0,0):: CALL L
OCATE(#1,X,Y-8):: GOTO 410
560 CALL GCHAR(X1,Y1+1,CH):: IF
CH=40 THEN 410
570 SCORE=SCORE+10
580 IF CH=34 THEN FLAG=-1 ELSE 6
00
590 FOR A=0 TO 30 STEP 10 :: CAL
L SOUND(-99,880,A):: NEXT A
600 CALL MOTION(#1,0,50)
610 CALL MOTION(#1,0,0):: CALL L
OCATE(#1,X,Y+8):: GOTO 410
620 CALL GCHAR(X1-3,Y1,CH):: IF
CH=40 THEN 410
630 SCORE=SCORE+10
640 CALL MOTION(#1,-100,0)
650 CALL MOTION(#1,0,0):: CALL L
OCATE(#1,X-32,Y):: GOTO 410

```



```

660-ERASE GCHAR(X1+1,Y1;CH); IF
CH>39 THEN 410
670 SCORE=SCORE+10
680 CALL MOTION(#1,100,0)
690 CALL MOTION(#1,0,0):: CALL L
OCATE(#1,X+32,Y):: GOTO 410
700 CALL COLOR(#1,16):: LIVES=L
IVES+1 :: FLAG=0 :: CALL SOUND(40
0,2000,0):: FOR A=1 TO 100 :: NE
XT A
710 DISPLAY AT(2,27):USING "##":
LIVES #: IF LIVES=3 THEN 720 ELS
E 290
720 HISCORE=MAX(SCORE,HISCORE)::
DISPLAY AT(1,12)SIZE(8):USING "
#####":SCORE :: DISPLAY AT(2,
12)SIZE(8):USING "#####":HISC
ORE
730 FOR A=0 TO 30 :: CALL SOUND(
-99,131,A):: CALL SOUND(-99,110
,A):: NEXT A :: DISPLAY AT(23,8)S
IZE(13):"((GAME OVER(("
740 DISPLAY AT(24,9)SIZE(11):"PR
ESS(AKEY"
750 CALL KEY(O,K,S):: IF S=0 THE
N 750 ELSE CALL HCHAR(23,1,40,64
):: GOTO 270
760 SUB INSTRUCTIONS
770 CALL SCREEN(5)
780 CALL CLEAR :: DISPLAY AT(2,7
):"HAUNTED(HOUSE" :: DISPLAY AT(
3,7):"(((((((((((((" :: DISPLAY
AT(5,1):"MEET FRED ;="
790 DISPLAY AT(6,1):"THE FROG
<>" :: DISPLAY AT(8,1):"FRED D
ECIDED TO STAY IN A QUANT LIT
TLE MANSION OVER NIGHT:"

```

```

800 DISPLAY AT(11,1):"BUT THIS W
AS NO ORDINARY HOME BUT THE H
AUNTED HOUSE [GASP!!]:Now FRED
IS RELYING ON YOU TO FREE HIM:"
810 DISPLAY AT(15,1):"BUT DON'T
LET HIM TOUCH ANY GHOSTS OR ELSE
YOU WILL SCARE FRED AND 3 S
CARES MEANS FRED WILL GO MAD
:"
820 CALL HCHAR(1,1,40,32):: CALL
VCHAR(1,1,40,21):: CALL VCHAR(1
,32,40,21):: CALL HCHAR(21,1,40,
32)
830 INPUT "PRESS ENTER":A$
840 DISPLAY AT(2,1)ERASE ALL:"TH
ERE ARE 15 SCREENS EACH TIME T
HE GHOSTS GETTING FASTER:"
850 DISPLAY AT(5,1):"TO GET FROM
ONE SCREEN TO ANOTHER YOU MUS
T GET A KEY [TOP RIGHT HAND COR
NER]"
860 DISPLAY AT(8,1):"AND OPEN TH
E GREEN DOOR [BOTTOM LEFT CO
RNER]:THIS IS ALL DONE BY TOUCHIN
G THE KEY AND DOOR:"
870 DISPLAY AT(12,1):"YOU MOVE F
RED BY THE ARROW KEYS [E S D X]
:"
GOOD LUCK
"
880 CALL HCHAR(1,1,40,32):: CALL
VCHAR(1,1,40,17):: CALL VCHAR(1
,32,40,17):: CALL HCHAR(17,1,40,
32)
890 INPUT "PRESS ENTER":A$
900 CALL CLEAR :: CALL SCREEN(2)
910 SUBEND

```



Ghost freak

I own a Spectrum and I love it! Right then, now I've got that over I can get on!

I'm writing about software piracy. I don't see what all the fuss is about. Why is it against the law? We copy records on to tapes all the time and programs off the TV and no-one seems to mind that.

I must admit that I copy the odd game or two and I could name 50 other people that do the same.

I want to thank a couple of people. Firstly Mastertronic for bringing out the £1.99 games range. Secondly, I would like to thank British Telecom for the Firebird range. When I go into software retailers I look for any good cheap games. I picked up Booty by Firebird and when I ran through it on my computer — wow! the graphics! the sound effects! It's not every day you find games like this, especially at £2.50.

Finally I would like to thank Activision for releasing Ghostbusters. After reading the book, playing the record, watching the film, reading the book again, and then buying the game I am really a Ghostbusters addict. But please could someone tell me why the price is not right. It is very wrong. Why, why does it have to be priced at £9.99?

Kevin Ringrose, Brixworth

At present there is a bill going through parliament which hopefully will clarify the legal position of software piracy. Surely the reason why software piracy should be illegal is because people spend months writing and producing programs, so it's only fair to pay them for all their hard work.

Joystick option

I'm writing to help any TI-99/4A owners who would like to play Hunchback Havoc by Lantern software, but can't because they don't own joysticks. Typing in the following lines will convert the program for use with the arrow keys.

```
1060 CALL KEY(),K,S)
1090 IF K=101 THEN 1140
1100 IF K=120 THEN 1530
1110 IF K=100 THEN 1360
1120 IF K=115 THEN 1720
```

Now the game can be played with the arrow keys, but make sure the alpha-lock is in the up position.

Also, if anyone is having trouble getting through the sheets on Hunchback Havoc, type in the following line:

```
1070 T=T-1
```

Now, the longer it takes you to put out the fires, the more time you will have to collect the jewels and go on to the next sheet.

Jon Paul McCalmont, Brecon

Make it better

With reference to the ZX81 program, Full Screen Editor, in HCW 97, may I point out a possible error. Perhaps it's just that it's slightly blurred in my copy, but line 230 has "Let cursor+Dfile +L+33 +C+1" when there should be a * instead of the + before 33.

May I also suggest an enhancement to enable the cursor to go automatically to the start of the next line when it reaches the end of the line above:

```
375 IF C=D32 THEN LET L=L+1
```

Finally, many thanks for continuing to support the ZX81.

L W Oliver, Perth

Mine's best

I am an 11-year-old owner of a TI-99/4A which I have had for two years.

At my school nearly all my mates own a computer. Five own Spectrums, one owns a C64, one an Amstrad, one a C16, three have BBCs, Electronics, TIs and VIC-20s.

My best mate boasts that because the Spectrum has the most games it is the best computer. But who cares when the TI has games like Buck Rogers, Indoor Soccer and many more? Maybe the Spectrum and C64 are best sellers, but three TI owners out of one class isn't bad considering the popularity of the 64.

While I'm writing I'd like to thank HCW staff for producing a magazine that covers all these machines at low cost compared to many others.

Mark Taylor, Leigh

Easy to program

I have just obtained a C16 and am very pleased with its performance and the ease with which I can program it. For the past three weeks I have bought HCW which I find interesting and very useful as it contains a few advertisements for software for the C16.

However, I would like to see a few C16 programs in HCW's curriculum. As, on the whole I enjoy HCW very much, please could you try and include programs for the growing number of C16 owners in the country.

Kevin O'Connor, Airdrie

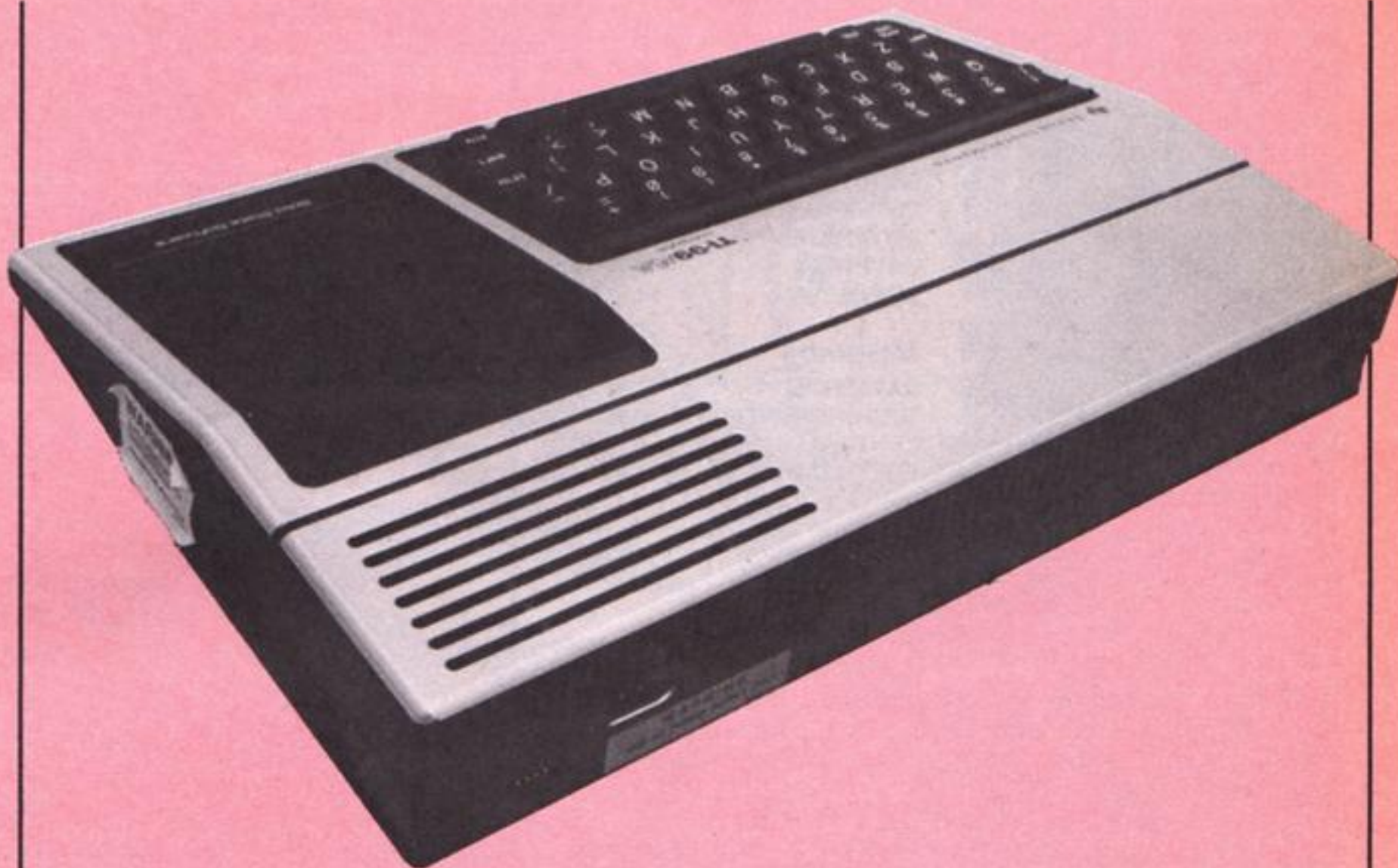
Thanks for your praise, Kevin. In HCW 101 we featured our first C16 program and hope to continue supporting this machine. However, we rely on readers' submissions for programs, so we need C16 owners to send their programs to us.

Home Computing Weekly,
No 1 Golden Square,
London W1R 3AB.

LETTERS PAGE

Joe Berry shows you how to program your Texas computer with the aid of the CTRL key. It's easy once you know how

HOW TO PROGRAM YOUR TI



In his excellent book *Mastering the TI-99* published by Micro Press, Peter Brooks, describing quirks specific to the TI computer, writes... "if you type a line number followed by REM and then hold the CTRL key down and press a few others followed by ENTER, and then attempt to list the line, you will find a whole string of characters and TI BASIC words in the REM line."

Successfully completing this experiment on line 100, I then tried typing (line 110), with the CTRL key held down, the (invisible!) vowel letters A E I O U, not forgetting to space the letters as usual. This was a REM line as before.

After pressing ENTER and LIST the following words appeared on the screen: REM ELSE GO DEF UNBREAK RANDOMIZE. I next typed, in the same method, the letter A (line 120) ELSE appeared after entering and listing. On line 130, U, as expected, came up with RANDOMIZE. Thus it seemed I could go through the alphabet, which proved to be the case, as follows:

A=ELSE, B=::, C=!, D=IF, E=GO, F=GOTO, G=GO-SUB, H=RETURN, I=DEF, J=DIM, K=END, L=FOR, M=LET, N=BREAK, N=BREAK, O=UNBREAK, P=TRACE, Q=UNTRACE, R=INPUT, S=DATA, T=RESTORE, U=RANDOMIZE, V

=NEXT, W=READ, X=STOP, Y=DELETE, Z=REM 1=TO, 2=STEP, 3=, (comma), 4=;, 5=;, 6=), 7=(, 8=OPTION, 9=OPEN, 0=THEN.

Taking the punctuation characters one finds that the semicolon (;) is the code for that important command PRINT, =ON, +=CALL, while strangely, perhaps, the comma =02 "+(!) Other BASIC WORDS like XOR, NOT, OR etc. can be found using the FCTN key in the same way.

All this of course, must be known to many, but to me — and others I hope — it's an interesting discovery. Could these letters and characters be used as a kind of shorthand code for programming, I wondered? The REM lines seemed to make this an impossibility, but I found that when running trial lines with the Extended BASIC module, one did not have to REM for single character conversion.

If you have Extended BASIC try the following two-line trial program, remembering to hold down the CTRL key while pressing the comma, adding a space then typing in the phrase enclosed in quotes.

```
100 ; 'PRESS ENTER AND
    RUN THIS LINE'
110 ; 'NOW PRESS ENTER
    AND LIST'
```

You will find that in the

listing the semicolons are replaced by PRINT, but the lines in longer programs need not be listed until it is convenient.

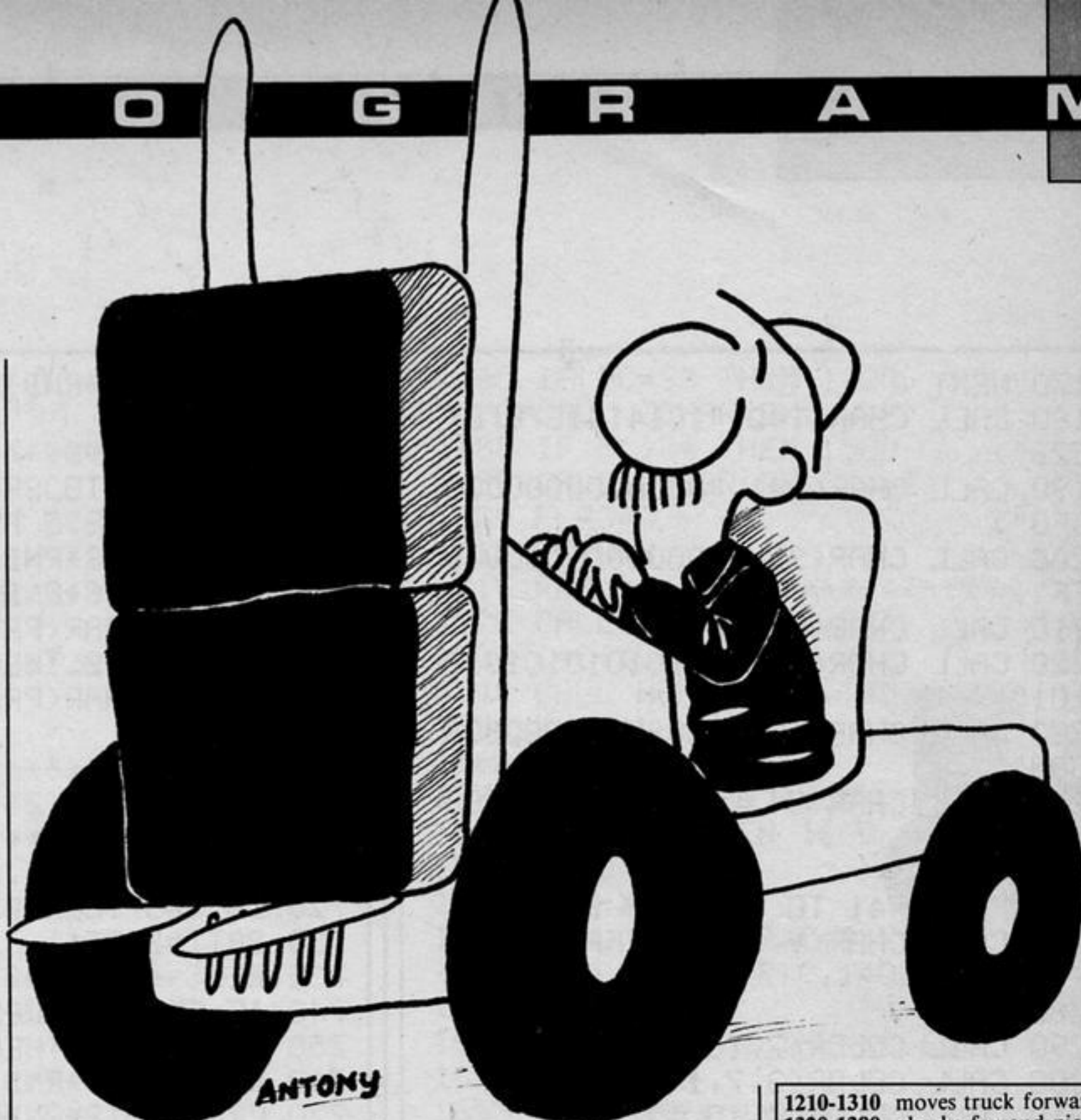
Now try this longer CTRL key program:

```
100 P
110 ; 'THE CODE FOR
    'TRACE' WAS THE
    LETTER P WITH THE
    CTRL KEY DOWN'
120 ; 'THE CODE FOR
    'PRINT' IS ;'
140 ; 'THIS WILL LIST AND
    RUN'
150 R 'THIS LINE WAS
    INPUT WITH THE
    LETTER 'R'. PRESS
    ENTER':AS$
160 Q
170 ;
180 ; 'NOTE THAT THE
    CODE FOR 'UNTRACE'
    (LINE 160) IS Q'
190 ; 'ALL LISTINGS ARE
    DISPLAYED CORRECT-
    LY, BUT REMEMBER
    THAT WHILE HOLDING
    THE CTRL: KEY DOWN
    THE CODE CHARACTER
    IS NOT VISIBLE!'
```

Is any of this of practical use, you may ask? I suggest that as one becomes conversant with the codes it should speed up programming, but in any case discovering as much as possible about the TI-99/4A will lead to a better understanding of the machine.

You're in charge of a forklift truck with failing batteries, in this game by Rod Lane in TI BASIC

KEEP ON TRUCKING



In this game you play the part of a forklift truck operator putting in some overtime at the warehouse. Unfortunately the only truck around is old and the battery in it doesn't last long! So to keep going you will probably need to recharge a number of times.

However the recharge points (shown as new batteries) keep appearing and disappearing. You should therefore only go for one if it appears near to you! To use these recharge points you must BACK your truck over them. If you are successful you will gain between one and four units of power (shown on the power level meter at the bottom of the screen).

The normal four cursor keys control the movement of the truck. Your job then, assuming your power lasts, is to pick up all the coloured crates by sliding the forks of your trucks into the pallet from the FRONT and to load them into the corresponding containers at the right of the screen.

When you have loaded the last crate and backed your truck out of the container the game ends and offers you a new game at the same or a different skill level. This skill level determines the number of crates to be loaded. Levels one and two are best only for practice. Speed is not essential as you use no power when standing still, so

plan your routes to conserve as much power as possible.

I have written the program with REM statements purely as dividing lines to help in understanding the program and as these REM statements are on line numbers ending in five, if you use the NUM command when typing the program in, you can ignore all the REMs. Good luck!

How it works

150-320 defines characters
330-700 sets up screen
710-730 scans keyboard, counts number of movements
740-810 checks for recharge point: if none, places one, 20% of the time
820-860 decides on direction of movement required
870-950 moves truck down
960-1040 moves truck up
1050-1120 moves truck back
1130 checks if truck backs over recharge point and acts accordingly

1210-1310 moves truck forward
1320-1380 checks for and picks up crate if present in front of truck
1390-1510 main subroutine common to all four moves, decreases the power
1520-1650 loads crates on to containers, keeps tally
1660-1720 win routine
1730-1780 lose routine
1790-1850 offers replay
1860-1880 skill level subroutine

Variables

R,C row, column position of truck
RX,CX row, column position of recharge point
PR,PC row, column position for placing crates
Q,V,D,C,T,Z all used in FOR-NEXT loops
FLAG if recharge point present
G,GN,GF used in GCHAR calls
BATT increments every move
BC position of power level on meter
X ASCII code of character on front of truck (fork or crate)
TOT total number of crates loaded
SK skill level

```
100 CALL CLEAR
110 GOSUB 1860
120 CALL CLEAR
130 RX=1
140 CX=1
145 REM *****
150 FOR Q=40 TO 64 STEP 8
160 CALL CHAR(Q,"78484878484878F
8")
```



```

170 NEXT Q
180 CALL CHAR(140,"1C14141E7E7E7E2B")
190 CALL CHAR(141,"00000000000000000F0")
200 CALL CHAR(95,"00000000000000000FF")
210 CALL CHAR(96,"FF")
220 CALL CHAR(105,"01010101010101010101")
230 CALL CHAR(106,"80808080808080808080")
240 CALL CHAR(34,"6666FF99FFFF99FF")
250 V=41
260 FOR V=41 TO 65 STEP 8
270 CALL CHAR(V,"FFFFFFFFFFFFFFFFF")
280 NEXT V
290 CALL COLOR(2,16,1)
300 CALL COLOR(3,7,1)
310 CALL COLOR(4,15,1)
320 CALL COLOR(5,2,1)
325 REM *****
330 PRINT
340 RANDOMIZE
350 BC=1
360 BATT=1
370 TOT=0
380 PRINT "    ----
    "
390 FOR P=1 TO 19
400 PRINT "    I
    J"
410 NEXT P
420 PRINT "    ~~~~~~
    "
430 PRINT "          POWER LEVEL"
440 W=5
450 N=27
460 FOR D=1 TO 2
470 FOR L=1 TO 2
480 FOR V=41 TO 65 STEP 8
490 CALL HCHAR(W,N,V)
500 W=W+5
510 NEXT V
520 W=6
530 NEXT L
540 W=5
550 N=28
560 NEXT D
570 R=INT(18*RND)+4

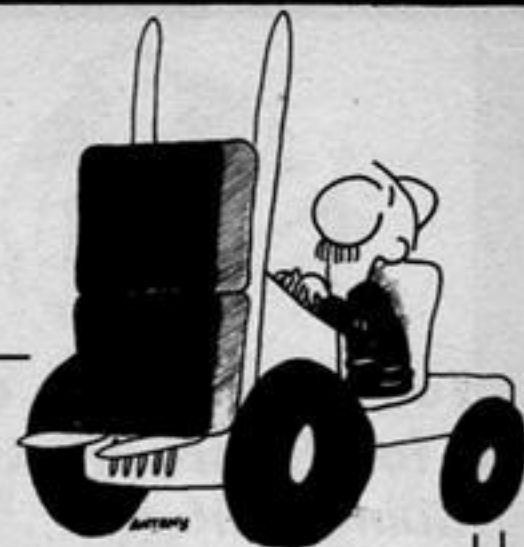
```



```

580 C=INT(20*RND)+6
590 X=141
600 GOSUB 1390
610 FOR T=1 TO SK
620 FOR Q=40 TO 64 STEP 8
630 PR=INT(18*RND)+4
640 PC=INT(18*RND)+8
650 CALL GCHAR(PR,PC,PG)
660 IF PG<>32 THEN 630
670 CALL HCHAR(PR,PC,Q)
680 NEXT Q
690 NEXT T
700 CALL HCHAR(24,1,41,32)
705 REM *****
710 CALL KEY(3,K,S)
720 IF S=0 THEN 710
730 BATT=BATT+1
735 REM *****
740 IF FLAG=1 THEN 820
750 IF RND<.8 THEN 820
760 RX=INT(13*RND)+7
770 CX=INT(13*RND)+8
780 CALL GCHAR(RX,CX,G)
790 IF G<>32 THEN 760
800 CALL HCHAR(RX,CX,34)
810 FLAG=1
815 REM *****
820 IF K=88 THEN 870
830 IF K=69 THEN 960
840 IF K=83 THEN 1050
850 IF K=68 THEN 1210
860 GOTO 710
865 REM *****
870 CALL GCHAR(R+1,C,G)
880 CALL GCHAR(R+1,C+1,GN)
890 IF (G=32)*(GN=32) THEN 900 ELSE 710
900 CALL HCHAR(R,C,32,2)
910 R=R+1
920 IF R<23 THEN 940
930 R=22
940 GOSUB 1390
950 GOTO 710
955 REM *****
960 CALL GCHAR(R-1,C,G)
970 CALL GCHAR(R-1,C+1,GN)
980 IF (G=32)*(GN=32) THEN 990 ELSE 710
990 CALL HCHAR(R,C,32,2)
1000 R=R-1
1010 IF R>3 THEN 1030
1020 R=4

```




```

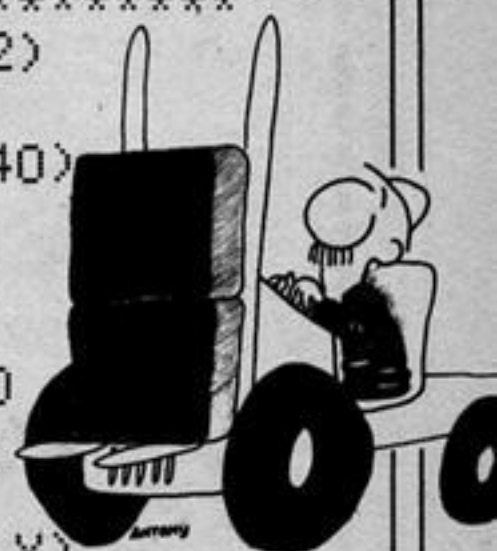
1030 GOSUB 1390
1040 GOTO 710
1045 REM *****
1050 CALL GCHAR(R,C-1,G)
1060 IF G=34 THEN 1080
1070 IF G<>32 THEN 710
1080 CALL HCHAR(R,C,32,2)
1090 C=C-1
1100 IF C>5 THEN 1120
1110 C=6
1120 GOSUB 1390
1125 REM *****
1130 IF G<>34 THEN 710
1140 E=INT(4*RND)+1
1150 BC=BC-E
1160 IF BC>0 THEN 1180
1170 BC=1
1180 CALL HCHAR(24,BC,41,32-BC)
1190 CALL SOUND(400,-8,2)
1200 GOTO 710
1205 REM *****
1210 CALL GCHAR(R,C+2,GF)
1220 IF (GF=106)+(GF=34) THEN 710
1230 IF GF=X+1 THEN 1520
1240 IF (GF=41)+(GF=49)+(GF=57)+(GF=65) THEN 710
1250 IF GF<>32 THEN 1320
1260 CALL HCHAR(R,C,32)
1270 C=C+1
1280 IF C<27 THEN 1300
1290 C=26
1300 GOSUB 1390
1310 GOTO 710
1315 REM *****
1320 IF X<>141 THEN 710
1330 X=GF
1340 CALL HCHAR(R,C,32)
1350 C=C+1
1360 GOSUB 1390
1370 CALL SOUND(-100,110,4,-7,16)
1380 GOTO 710
1385 REM *****
1390 CALL SOUND(-200,-7,12)
1400 CALL HCHAR(R,C,140)
1410 CALL HCHAR(R,C+1,X)
1420 IF BATT/10<>INT(BATT/10) THEN 1510
1430 CALL SOUND(250,550-(13*BC),5)
1440 CALL HCHAR(24,BC,49)
1450 BC=BC+1

```

```

1460 IF BC=33 THEN 1730
1470 CALL GCHAR(RX,CX,F)
1480 IF F<>34 THEN 1500
1490 CALL HCHAR(RX,CX,32)
1500 FLAG=0
1510 RETURN
1515 REM *****
1520 CALL HCHAR(R,C,32)
1530 C=C+1
1540 CALL HCHAR(R,C,140)
1550 X=141
1560 CALL KEY(O,K,S)
1570 IF S=0 THEN 1560
1580 IF K=83 THEN 1600
1590 GOTO 1560
1600 C=C-1
1610 CALL HCHAR(R,C+1,X)
1620 CALL HCHAR(R,C,140)
1630 TOT=TOT+1
1640 IF TOT=SK*4 THEN 1660
1650 GOTO 710
1655 REM *****
1660 FOR Z=1 TO 24
1670 PRINT "FINISHED FINISHED FINISHED"
1680 NEXT Z
1690 FOR Z=1 TO 24
1700 PRINT " "
1710 NEXT Z
1720 GOTO 1790
1725 REM *****
1730 FOR Z=1 TO 24
1740 PRINT "NO POWER NO POWER NO POWER"
1750 NEXT Z
1760 FOR Z=1 TO 24
1770 PRINT " "
1780 NEXT Z
1785 REM *****
1790 INPUT "MORE??? [Y/N] " : AS$
1800 IF AS$<>"N" THEN 1830
1810 CALL CLEAR
1820 END
1830 IF AS$<>"Y" THEN 1790
1840 GOSUB 1860
1850 GOTO 330
1855 REM *****
1860 INPUT "SKILL LEVEL? [UP TO 4] " : SK
1870 IF (SK<>1)*(SK<>2)*(SK<>3)*(SK<>4) THEN 1860
1880 RETURN

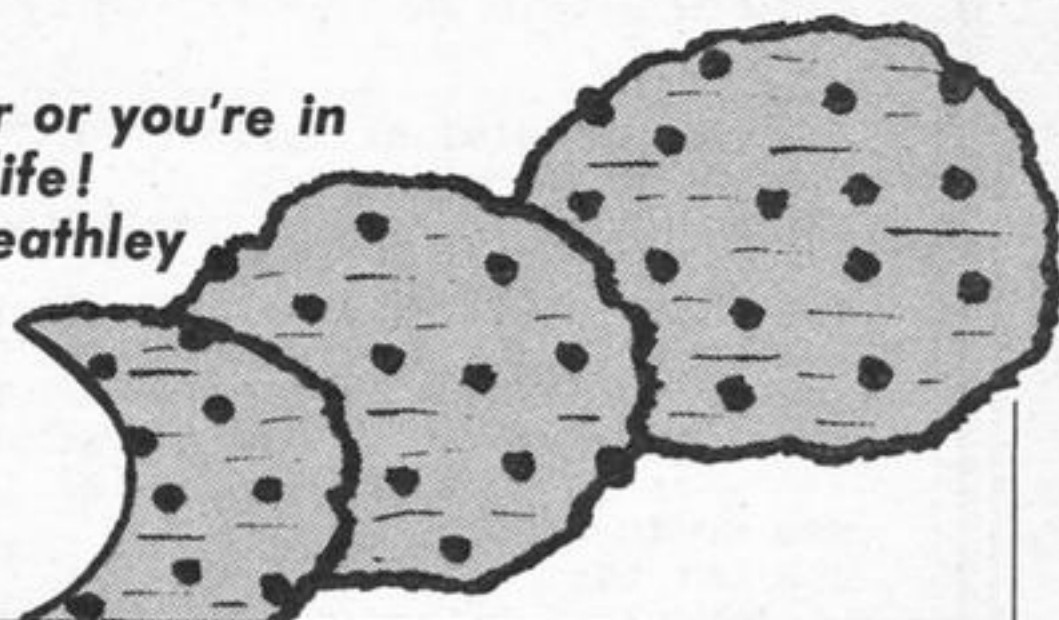
```



COOKIE MONSTER



Avoid the cookie monster or you're in danger of your life!
By Paul and Cara Leathley



The object of this game is to collect as many cookies as possible, avoiding the cookie monster, who is also eating them. Ladders are provided to enable you to move from one floor to the next.

You lose your life if you collide with the cookie monster or touch a ledge. If all the cookies are eaten then the game continues with a different screen and new cookies.

Use F and H to go left and right respectively and L to go up ladders. There are holes in the ledges, down which you may drop with loss of life, by walking off the edge.

How it works

- 10-160 defines characters and colours
- 190-240 gets light green screen and prints "cookie" in large letters at top of screen
- 270-360 draws ledges
- 370-460 prints ladders
- 470-560 prints cookies
- 570-630 sets initial values of variables
- 700-750 checks which key, if any, is pressed
- 760-790 changes shape of man as he turns right or left
- 800 checks man doesn't leave screen
- 810-860 checks what man hits next
- 870-920 checks whether man has gone off edge of ledge
- 940-1010 man goes up
- 1030-1040 changes shape of man as he climbs ladder
- 1050-1070 touch of ledge
- 1080-1100 collision with cookie monster
- 1110-1210 man hits cookie
- 1230-1280 new game option
- 1290-1580 moves monster
- 1590-1610 prints scores in terms of cookies

Variables

- SCM player's score
- SCC monster's score
- MX, MY co-ordinates of man
- CX, CY co-ordinates of cookie monster
- CHR character number for man: 96 = left, 97 = right, 98 = up ladder
- PCM character number of what man last hit
- PCC character number of what cookie monster last hit
- T number of man's moves since monster moved last
- MCX how many spaces monster moves vertically
- MCY how many spaces monster moves horizontally

```

10 CALL CHAR(96,"181838FE181C744
6")
20 CALL CHAR(97,"18181C7F18382E6
2")
30 CALL CHAR(98,"5A5A7E183C2466"
)
40 CALL COLOR(9,2,4)
50 CALL CHAR(104,"81FF818181FF81
81")
60 CALL COLOR(10,11,4)
70 CALL CHAR(112,"FFFF44AAAA44FF
FF")
80 CALL COLOR(11,5,4)
90 CALL CHAR(120,"007E5AFF242466
")
100 CALL CHAR(121,"FFFFFFFFFFFFFF
FFF")
110 CALL CHAR(122,"FFFFFFFF")
120 CALL CHAR(123,"00000000FFFFF
FFF")
130 CALL CHAR(124,"0000FFFFFFFFFF
000")
140 CALL COLOR(12,14,4)
150 CALL CHAR(128,"3C42A98391C34
A3C")
160 CALL COLOR(13,7,4)
170 SCM=0
180 SCC=0
190 CALL CLEAR
200 CALL SCREEN(4)
210 PRINT "      YZZ YZY YZY Y Y Y
YZZ"
220 PRINT "      Y      YXY YXY YY Y
Y!!"
230 PRINT "      Y(( Y(Y Y(Y Y Y Y
Y((("
240 PRINT "::::::::::::::::::::::::::"
250 C$(0)="COOKIE"
260 C$(1)="COOKIES"
270 FOR I=7 TO 23 STEP 2
280 RANDOMIZE
290 A=INT(RND*28)+3
    
```



```

300 B=INT(RND*10)
310 IF 30-A<B THEN 280
320 CALL HCHAR(I,3,112,28)
330 CALL HCHAR(I,A,32,B)
340 NEXT I
350 CALL HCHAR(23,3,112,28)
360 CALL HCHAR(5,3,112,28)
370 FOR I=8 TO 22 STEP 2
380 RANDOMIZE
390 P(1)=INT(RND*28)+3
400 P(2)=INT(RND*28)+3
410 P(3)=INT(RND*28)+3
420 FOR C=1 TO 3
430 CALL HCHAR(I,P(C),104)
440 CALL HCHAR(I-1,P(C),104)
450 NEXT C
460 NEXT I
470 FOR I=6 TO 22 STEP 2
480 FOR C=1 TO 3
490 RANDOMIZE
500 P(C)=INT(RND*28)+3
510 IF ((I=6)*(P(C)=3))+((I=22)*
(P(C)=3)) THEN 490
520 CALL GCHAR(I,P(C),GET)
530 IF (GET=104)+(GET=128) THEN 4
90
540 CALL HCHAR(I,P(C),128)
550 NEXT C
560 NEXT I
570 MX=22
580 MY=3
590 CHR=96
600 PCM=32
610 CX=6
620 CY=3
630 PCC=32
640 T=T+1
650 CALL HCHAR(MX,MY,CHR)
660 CALL HCHAR(CX,CY,120)
670 IF T<=1 THEN 690
680 GOSUB 1290
690 TIM=0
700 CALL KEY(3,K,S)
710 IF S<>0 THEN 740
720 TIM=TIM+1
730 IF TIM=20 THEN 680 ELSE 700
740 IF K=76 THEN 940
750 IF (K<>70)*(K<>72) THEN 700
760 ON K/2-34 GOTO 770,790
770 CHR=97
780 GOTO 800
790 CHR=96

```

```

800 IF (MY+(K-71)<3)+(MY+(K-71)>
30) THEN 700
810 CALL GCHAR(MX,MY+(K-71),Z)
820 IF Z<>32 THEN 930
830 MY=MY+(K-71)
840 CALL HCHAR(MX,MY-(K-71),PCM)
850 PCM=Z
860 IF PCM=104 THEN 640
870 CALL GCHAR(MX+1,MY,CH)
880 IF CH<>32 THEN 640
890 MX=MX+1
900 CALL HCHAR(MX-1,MY,PCM)
910 CALL HCHAR(MX,MY,CHR)
920 GOTO 870
930 ON (Z/8)-12 GOTO 1030,1050,1
080,1110
940 CALL GCHAR(MX-1,MY,Z)
950 IF Z<>32 THEN 980
960 IF PCM<>104 THEN 700
970 GOTO 990
980 IF Z<>104 THEN 1020
990 MX=MX-1
1000 CALL HCHAR(MX+1,MY,PCM)
1010 GOTO 850
1020 ON (Z/8)-13 GOTO 1050,1080,
1110
1030 CHR=98
1040 GOTO 830
1050 CALL CLEAR
1060 PRINT "OUCH! YOU BUMPED INT
O THE ":" LEDGE!"
1070 GOTO 1220
1080 CALL CLEAR
1090 PRINT "COOKIE MONSTER ATE Y
OU!"
1100 GOTO 1220
1110 CALL SOUND(500,880,0)
1120 SCM=SCM+1
1130 IF SCM+SCC=27 THEN 190
1140 IF (PCM<>104)+(K<>76) THEN 1
180
1150 MX=MX-1
1160 CALL HCHAR(MX+1,MY,PCM)
1170 GOTO 640
1180 MY=MY+(K-71)
1190 CALL HCHAR(MX,MY-(K-71),PCM
)
1200 PCM=32
1210 GOTO 870
1220 GOSUB 1590
1230 PRINT ":" PLAY AGAIN ?(Y
/N)"

```



```

1240 CALL KEY(3,K,S)
1250 IF S=0 THEN 1240
1260 IF K=78 THEN 1280
1270 GOTO 170
1280 END
1290 IF CX<>MX THEN 1310
1300 MCX=0
1310 IF CX<=MX THEN 1330
1320 MCX=-1
1330 IF CX>=MX THEN 1350
1340 MCX=1
1350 IF CY<>MY THEN 1370
1360 MCY=0
1370 IF CY<=MY THEN 1390
1380 MCY=-1
1390 IF CY>=MY THEN 1410
1400 MCY=1
1410 CALL GCHAR(CX+MCX,CY+MCY,CH
A)
1420 IF (CHA<>32)*(CHA<>104)*(CH
A<>112) THEN 1500
1430 CX=CX+MCX
1440 CY=CY+MCY

```

```

1450 CALL HCHAR(CX-MCX,CY-MCY,PC
C)
1460 PCC=CHA
1470 CALL HCHAR(CX,CY,120)
1480 T=0
1490 RETURN
1500 IF (CHA<96)+(CHA>98) THEN 15
10 ELSE 1080
1510 CALL SOUND(500,220,0)
1520 SCC=SCC+1
1530 IF SCC+SCM=27 THEN 190
1540 CX=CX+MCX
1550 CY=CY+MCY
1560 CALL HCHAR(CX-MCX,CY-MCY,PC
C)
1570 PCC=32
1580 GOTO 1470
1590 PRINT : "COOKIE MONSTER GOT"
: SCC; C$(SCC=1)+1)
1600 PRINT : "YOU GOT"; SCM; C$(S
CM=1)+1)
1610 RETURN

```

It's easy to complain about advertisements. But which ones?

Every week millions of advertisements appear in print, on posters or in the cinema.

Most of them comply with the rules contained in the British Code of Advertising Practice.

But some of them break the rules and warrant your complaints.

If you're not sure about which ones they are, however, drop us a line and we'll send you an abridged copy of the Advertising Code.

Then, if an advertisement bothers you, you'll be justified in bothering us.

The Advertising Standards Authority.
If an advertisement is wrong, we're here to put it right. ✓

ASA Ltd, Dept 2 Brook House, Torrington Place, London WC1E 7HN

This space is donated in the interests of high standards of advertising.

HCW needs you

Have you ever looked at the articles and programs in HCW and thought you could have written that?

Why not try? We welcome submissions from readers and are always interested in your ideas and suggestions.

Submissions tend to fall into one of three categories...

Programs are always supplied on cassette and are accompanied by full details of the program variables. Please type these details double-spaced. Listings are helpful, but not essential. What is vital is that the programs should be completely error free, so please double check.

Articles on certain aspects of using home computers should be no longer than 1000 words. Try to keep to the style you see in HCW and include programming examples where they will help the reader understand the subject. We will convert your sketched illustrations into final artwork.

We also use **short tips**, articles and **brief programming routines**. Any discoveries you have made about your machine might be of interest to other readers.

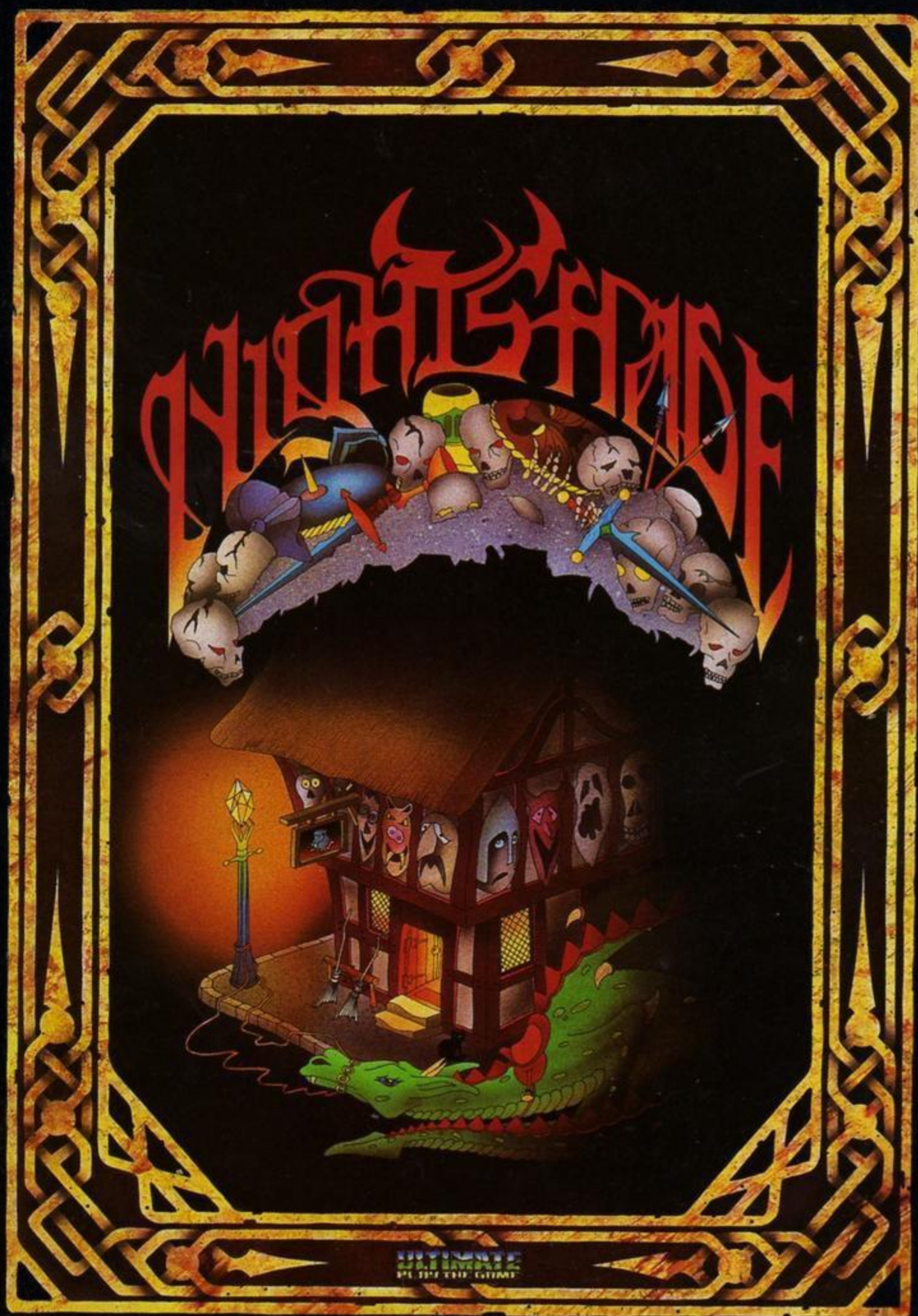
All contributions are acknowledged and those accepted for publication are paid for at competitive rates. The copyright in such work will pass to Argus Specialist Publications Ltd. Keep a copy of your work and include a telephone number and an SAE. Please label everything clearly with your name and the machine concerned.

All contributions should be sent to:

Submissions

Home Computing Weekly

No.1 Golden Square London W1R 3AB



**This collection is for all TI99/4A user.
If you like TI99 join International Forum
present on AtariAge site.
Greetings from Italy.**

Astharot